

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Návrh a realizace privátní IP telefonní sítě na bázi open source
ústředny

Design and Realization Private IP Telephone Network Based
on Open Source Exchange

2010

Michal Mátl

Prohlášení

„Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.“

V Ostravě dne 27. dubna 2010

.....

Poděkování

Rád bych na tomto místě poděkoval Ing. Pavlu Nevludovi a Ing. Pavlu Zubkovi ze střední školy telekomunikační za jejich vedení a připomínky, které mi pomohly vypracovat bakalářskou práci. Dále bych zde chtěl poděkovat Jaroslavu Šimkovi za poskytnutí potřebného technického vybavení, které bylo nutné pro vypracování mé práce.

Abstrakt

Hlavní náplní této práce je realizace telefonní ústředny na bázi Open Source projektu Asterisk. Jako operační systém jsem zvolil Linux kvůli jeho bezpečnosti a stabilitě. Cílem je vytvořit kompletně funkční ústřednu s vyřešeným Dial plánem a tarifací. Poté bude následovat konfigurace jednotlivých koncových zařízení. V úvodní části práce rozeberu principy a zpracování hlasu IP telefonie. Dále budu vysvětlovat, jak fungují nejpoužívanější protokoly SIP a H.323. V poslední části vytvořím návod, jak si udělat krok po kroku svoji privátní IP ústřednu.

Klíčová slova: Elastix, IP PBX ústředna, CentOS, GNU/Linux, ISDN, SIP Trunk, H.323, Asterisk, FreePBX, OpenVox, Tarifkace, VOIP

Abstract

The main content of this thesis is an implementation of telephone exchange based on Open Source project Asterisk. As an OS I choosed Linux for its security and stability. The aim is to create a completely functional PBX with a Dial plan and pricing. The next step is a configuration of each end point. In an introductory part of my thesis I will analyse principles and IP telephony voice processing. Then I am going to explain how the most popular protocols SIP and H.323 work. In the last part of the thesis I will describe how to create own exchange steb by step.

Keywords: Elastix, IP PBX Exchange, CentOS, GNU/Linux, ISDN, SIP Trunk, H.323, Asterisk, FreePBX, OpenVox, Pricing, VOIP

Seznam použitých zkratek

ACELP	-	Algebraic Code Excited Linear Prediction
ADPCM	-	Adaptive differential Pulse Code Modulation
CELP	-	Code Excited Linear Prediction
C RTP	-	Compressed Real-Time Protocol
CS-ACELP	-	Conjuncture Structure Algebraic Code Excited Linear Prediction
EDGE	-	Enhanced Data Rates for Global Evolution
GSM	-	Global system for mobile Communications
HDD	-	Hard Disk Drive
HTTP	-	Hyper Text Transfer Protocol
iLBC	-	internet Low Bit Rate Codec
IP	-	Internet Protocol
ISDN	-	Integrated Services Digital Network
LD-CELP	-	Low Delay Code Excited Linear Prediction
LPC	-	Linear Predictive Coding
MCU	-	Multipoint Control Unit
PBX	-	Private Branch Exchange
PCM	-	Pulse Code Modulation
PER	-	Packet Encoding Rules
PSQM	-	Perceptual Speech Quality Measure
QOS	-	Quality of Service
RTP	-	Real Time Protocol
SDP	-	Session Description Protocol
SIP	-	Session Initiation Protocol
SMTP	-	Simple Mail Transfer Protocol
SRTP	-	Secure Real Time Protocol
SSH	-	Secure Shell
TCP	-	Transmission Control Protocol
UDP	-	User Datagram Protocol
UMTS	-	Universal Mobile Telecommunications systém
URI	-	Uniform Resource Identifikator
VOIP	-	Voice over IP
XMPP	-	Extensible Messaging and Presence Protocol

Obsah

1. Úvod.....	1
2. Principy IP telefonie.....	2
2.1 Výhody a nevýhody.....	2
2.2 Základní typy VOIP.....	3
3. Zpracování hlasu.....	4
3.1 Kodeky	5
4. Protokol RTP	8
4.1 RTP RFC 1889.....	9
4.2 RFC 3550.....	10
5. Signalizační protokoly.....	11
5.1 H.323.....	11
5.1.1 Architektura H.323.....	11
5.1.2 Signalizace.....	13
5.1.3 Kodeky.....	15
5.2 SIP (Session Initiation Protocol).....	15
5.2.1 Adresace v SIP.....	15
5.2.2 Architektura SIP	15
5.2.3 Požadavky.....	17
5.2.4 Odpovědi.....	18
5.2.5 Transakce a dialogy.....	19
6. Realizace IP ústředny.....	21
6.1 Instalace.....	21
6.2 Konfigurace Elastixu.....	24
6.2.1 Informační tabule a změna hesla admin účtu	25
6.2.2 Instalace karty.....	26
6.2.3 Extensions (Volající uživatelé).....	27
6.2.4 SIP a ISDN Trunk.....	28
6.2.5 Tarifkace odchozích hovorů.....	31
7. Závěr.....	35
8. Použité zdroje.....	37
8.1 Internet.....	37
8.2 Literatura.....	38
Přílohy.....	39
Konfigurace klientů.....	39
Obsah přiloženého DVD.....	41

Seznam obrázků

Obr. 1 Spojení počítač/počítač.....	3
Obr. 2 Spojení počítač/telefon.....	3
Obr. 3 Spojení telefon/telefon.....	3
Obr. 4 Analogový signál.....	4
Obr. 5 Kódování a dekódování signálu.....	4
Obr. 6 PCM	6
Obr. 7 RTP zařazení do síťového modelu.....	8
Obr. 8 Hlavička RTP.....	9
Obr. 9 Vrstvy H.323.....	11
Obr. 10 Entity H.323.....	12
Obr. 11 Inicializace hovoru.....	13
Obr. 12 Navázání hovoru.....	14
Obr. 13 Typická komunikace H.245.....	14
Obr. 14 Role UAC a UAS.....	16
Obr. 15 Topologie sítě.....	17
Obr. 16 Žádost INVITE.....	17
Obr. 17 Transakce a dialog.....	19
Obr. 18 Registrace klienta na server.....	20
Obr. 19 Úvodní obrazovka instalace.....	23
Obr. 20 Výběr jazyka.....	23
Obr. 21 Rozdělení disku.....	24
Obr. 22 Výběr balíků.....	24
Obr. 23 Začátek instalace.....	24
Obr. 24 Konfigurace sítě.....	25
Obr. 25 Úvodní obrazovka do systému CentOS.....	25
Obr. 26 Chyba certifikátu.....	26
Obr. 27 Úvítací obrazovka webového rozhraní Elastixu.....	26
Obr. 28 Informační tabule po přihlášení.....	27
Obr. 29 Změna informací účtu administrátora.....	27
Obr. 30 Neregistrovaná ISDN karta.....	27
Obr. 31 Detekce hardwaru.....	28
Obr. 32 Aktivace kanálů.....	28
Obr. 33 Registrace karty.....	28
Obr. 34 Přidání SIP účtu.....	29
Obr. 35 Tlačítko pro restart ústředny a aplikování změn.....	29
Obr. 36 Výběr Trunku.....	30
Obr. 37 Vytvoření Trunku.....	30
Obr. 38 Přidání odchozího volání pro Trunk.....	31
Obr. 39 Příchozí hovory směrované na uživatele.....	31
Obr. 40 Konfigurace spojení na providera.....	32
Obr. 41 Registrační řetězec.....	32
Obr. 42 Všechny záznamy hovorů.....	33
Obr. 43 Tarifikační plány.....	33
Obr. 44 Vytvoření nového plánu.....	33
Obr. 45 Nastavení účtování.....	34
Obr. 46 Chyba v group=0,63.....	34
Obr. 47 Výpis hovorů s tarifikací.....	35
Obr. 48 Graf volaných minut v jednotlivých tarifikacích.....	35
Obr. 49 Graf jednotlivých tarifikací dle ceny.....	35

Obr. 50 Shrnutí volání uživatelů.....	36
Obr. 51 Top 10 odchozích a příchozích hovorů.....	36
Obr. 52 Přidání účtu.....	41
Obr. 53 Přihlašovací údaje.....	41
Obr. 54 Sipdroid - přidání účtu	42

1. Úvod

V dnešní době se internet dostává do více a více domácností a archaická technologie jako je ISDN se pomalu ruší. Nástupcem této technologie je IP telefonie. Tato technologie v posledních letech zaznamenala velký rozmach. Její nespornou výhodou je, že člověk může volat po celém světě za velice nízké náklady. Ve většině případů zaplatíte jen poplatek za internet a jakkoliv dlouhý hovor může být zadarmo. IP telefonie je služba, která umožňuje přenos hlasu přes IP protokol. Díky IP telefonii můžeme v jedné ústředně mít připojení na místní síť, ISDN linky a dokonce v dnešní době i na mobilní telefony.

A jak toto všechno funguje? To si objasníme v první kapitole. Tam budu rozebírat hlavní principy, jaké architektury se používají a jak se přenáší hlas po síti, kde funguje protokol IP. V druhé kapitole se podíváme na zpracování hlasu. Rozebereme si zde kódovací a dekódovací algoritmy, sloužící ke kompresi přenášených dat, což umožňuje snížit datový tok. Ve třetí kapitole se budu věnovat nejpoužívanějším protokolům, které dnes dominují v IP telefonii.

V praktické části budu realizovat telefonní ústřednu na bázi Asterisku a Freepbx. Navrhnou privátní řešení pro volání a také aby se telefonní ústředna zvládala napojit na veřejné komunikační síť. S návrhem na napojení do veřejných sítí souvisí také vyřešení tarifkace pro jednotlivé uživatele. Po vyřešení těchto problémů mě bude čekat konfigurace jednotlivých koncových zařízení pro komunikaci a pak následné vyzkoušení funkčnosti mého návrhu. Realizaci řádně zdokumentuji a vytvořím tak návod, na jehož základě si můžeme postavit svoji funkční ústřednu.

2. Principy IP telefonie

Abychom pochopili správně, jak funguje IP telefonie, musíme si povědět, jak fungují klasické telefonní linky. U nich se používá síť přepojovaných okruhů. Tento název přesně vystihuje, jak daná technologie funguje. Když jeden účastník zavolá druhému a hovor se spojí, dojde k vyhrazení jednoho kanálu pro daný hovor. A jakmile účastníci ukončí hovor, virtuální okruh se tím zruší a uvolní se pro další hovory. Tento způsob má samozřejmě své výhody i nevýhody. Na jednu stranu můžeme zajistit dobrou kvalitu hovoru, na stranu druhou ale dochází k ne hospodárnému využití datového pásma – nelze totiž využít nevyužitou kapacitu datového pásma pro jiný datový přenos.

U IP telefonie je ovšem princip úplně jiný. Zde se využívá techniky přepínání paketů. To znamená, že se na straně hovořícího účastníka datový tok rozdělí do malých bloků dat (paketů), a ty jsou následně vyslány sítí k druhému účastníkovi. U příjemce se náš datový tok rozdělený na pakety opět složí do původní podoby, a ve výsledku účastník slyší náš hlas. Při referenci ke standardnímu síťovému modelu a rozložení jednotlivých úloh přenosu do různých vrstev se pak ve skutečnosti u nejnižších úrovní vlastně jedná o data jako každá jiná, konverze hlasu na data a naopak probíhá ve vyšších vrstvách. Ovšem tento popis je velice hrubý, proto si v další kapitole si povíme podrobněji, v jaké podobě a jak přesně se náš hlas šíří datovou sítí.

2.1 Výhody a nevýhody

Výhoda tohoto principu je, že se může po jednom médiu šířit spolu s dalšími službami. Většinou totiž nebývá využita celá šířka datového pásma. A tak vedle přehrávání hudby, videa, stahování z internetu apod. můžeme i volat. Další z výhod je, že můžeme ušetřit nemalé peníze za volání. Hodně se dá ušetřit na dálkových hovorech, kde se cena u VOIP pohybuje v relacích jako při volání do místní sítě, někdy je cena dokonce mnohem nižší. V rámci privátní sítě je volání zadarmo. Dnes už tedy firmy nemusí investovat peníze do datové sítě – a zvláště do sítě pro běžné volání – ale vytvoří se jen datová síť, tím pádem mají k dispozici volání a datové přenosy v jednom. Jednodušší tak je např. přesun zaměstnance v rámci firmy nebo připojení nového pracovníka. Dříve bychom museli volat speciální firmu na kabeláže a složité překonfigurovat telefonní ústřednu. A to obnáší další finance.

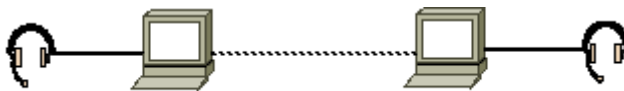
Abychom IP telefonii pouze nechválili, musím zde zmínit také nevýhody tohoto principu. Nároky na síť u VOIP jsou poměrně dost vysoké. Takže abychom mohli využívat hlas přes IP síť, musíme mít kvalitní linku, která má dobrou datovou propustnost a dobu odezvy. Je požadována odezva menší než 150 milisekund. Pokud máme nekvalitní připojení, dochází k výpadkům spojení, může se objevit i ozvěna a někdy i zkreslení hlasu. Další nevýhodou je bezpečnost volání. V počátcích VOIP nebyl brán ohled na odposlechnutí hovoru. V současné době se pracuje na verzi RTP protokolu, která má za úkol zajistit větší bezpečnost VOIP.

V budoucnu by snad už neměl být problém s provozováním IP telefonie, jelikož dochází neustále k navyšování kapacity sítě internetu. Zlepšují se rovněž kompresní techniky, které nám umožňují snížit datový tok a navíc se zavádí do IP sítě kvalita služby (Qos – quality of service), která má za úkol řídit datový tok v sítích a určovat prioritu jednotlivých paketů.

2.2 Základní typy VOIP

Hrubé rozdělení VOIP je dle koncových zařízení. Samozřejmě by se dalo najít spoustu parametrů podle kterých by se dala IP telefonie rozdělit. Ale pro naši potřebu jsem zvolil jen toto základní.

- Nejčastěji lidé používají softwarové telefony. Tzv. typ **Počítač/Počítač**. Tento způsob je nejrozšířenější díky své ceně. Většina softwarových telefonů je zadarmo. Uživatel musí mít pouze zvukovou kartu, mikrofon a sluchátka.



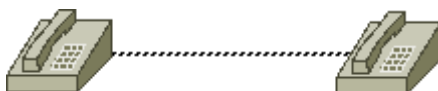
Obr. 1: Spojení počítač/počítač

- **Počítač/Telefon** – v dnešní době všichni poskytovatelé VOIP jsou napojeni na veřejnou telefonní síť a do mobilních sítí, takže není problém se dovolat ze svého počítače komukoliv.



Obr. 2: Spojení počítač/telefon

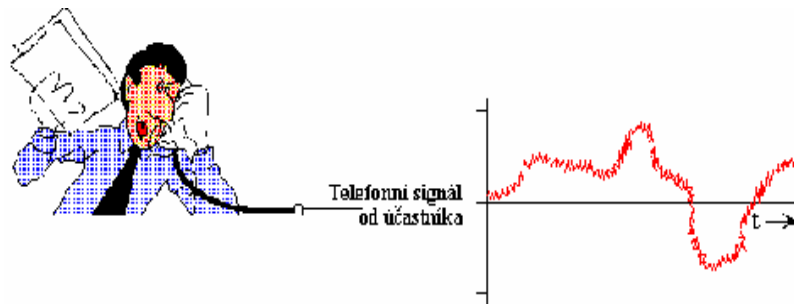
- **Telefon/Telefon** – v IP telefonii můžeme využívat jak klasické telefony, tak i speciální IP telefony. A to díky variabilitě, jež nám VOIP nabízí. Klasický telefon můžeme využít pomocí brány, která se postará o převod do digitální podoby, a následně data posílá na ústřednu našeho poskytovatele. U IP telefonů nepotřebujeme žádnou speciální bránu na převod hlasu, lze jej přímo zapojit do switchu nebo routeru. Výhoda použití hardwarových telefonů je nezávislost na spuštění počítače. Tím pádem můžeme být stále na příjmu.



Obr. 3: Spojení telefon/telefon

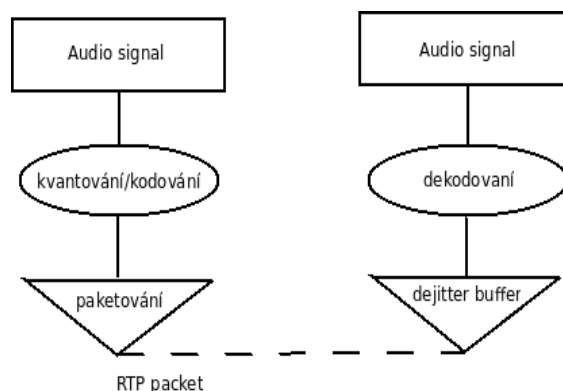
3. Zpracování hlasu

Když začneme mluvit do mikrofonu telefonu, náš hlas je v analogové podobě. Pro šíření přes IP síť jej potřebujeme dostat do digitální podoby. K tomu je nutné několik kroků než bude náš hlas vypuštěn do internetové sítě.



Obr. 4: Analogový signál

Takto nějak by vypadal náš hovor, pokud bychom použili analogový přístroj, a nepřeváděli hlas do digitální podoby. Tak to v praxi fungovalo až do roku 1972. Tehdy nastala změna v podobě standartu PCM, který přinesl do světa telekomunikací digitální podobu. Na následujícím obrázku vidíme princip zpracování hlasu .



Obr. 5: Kódování a dekódování signálu

Od účastníka jde audio signál, který je následně digitalizován a kódován. Kódování je závislé na použitém kodeku. Data, která vzniknou, se rozdělí na pakety s užitečnou zátěží 20 až 160 oktetů a se standardní hlavičkou 40 oktetů. Pak se pakety vyšlou v časovém rozestupu přes transportní RTP protokol druhé straně. U příjemce je přijatý paket zařazen do vyrovnávací paměti (dejitter buffer), která má za úkol vyrovnávat zpoždění hlasu. Z bufferu jsou posílány data do dekodéru a následně převedena na audio signál. Hlas v bitové podobě pak díky kodekům může mít velice malé požadavky na síť. Při použití technologie VOIP může stačit i 16 kbps. Při dodatečných úpravách dokonce pouhých 6 kbps.

3.1 Kodeky

Slovo kodek vzniklo ze slov kódovat a dekódovat. Kodeky mohou být řešeny strojově nebo softwarově algoritmem. Slouží k zakódování a ke kompresi analogového signálu, a následně k jeho rozkódování a k dekompresi na původní signál. Cílem kodeku je udělat co největší kompresi s co nejnížší ztrátou kvality přenášeného hovorového signálu. Čím méně bitů potřebujeme k zakódování informace, tím menší je požadavek na přenášené pásmo. V ideálním případě bychom neměli vůbec rozeznat kódovaný signál od nekódovaného. Ale jelikož kodeky často používají ztrátovou kompresi, dekódovaná data nejsou totožná s kódovanými.

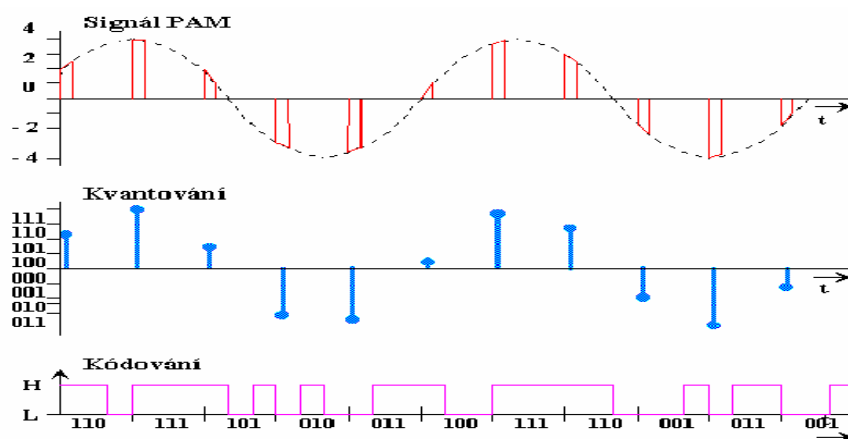
Seznam nepoužívanějších kodeků pro VOIP uvádí následující tabulka.

označení	standartizoval	Popis	Bit rate [kb/s]	Sampling rate [kHz]	Frame size [ms]	Mos
G.711	ITU-T	PCM	64	8	0,13	4,1
G.721	ITU-T	ADPCM	32	8		
G.722	ITU-T	ADPCM	48/56/64	16		
G.722.1	ITU-T	AMR-WB	24/32	16	20	
G.723.1	ITU-T	ACELP/MP-MLQ	5,33/6,4	8	30	3,8-3,9
G.726	ITU-T	ADPCM	16/24/32/40	8	10	3,85
G.728	ITU-T	LD-CELP	16	8	0,63	3,61
G.729	ITU-T	CS-ACELP	8	8	10	3,92
GSM	ETSI	RPE-LTP	13	8	22,5	3,27
SPEEX			8/16/32	2,15-24,6	30	
ILBC			8	13,3	30	3,81

Tab. 1: Nejpoužívanější kodeky

- **Bit Rate** je hodnota, která nám ukazuje, jak velkou potřebujeme datovou propustnost
- **Sampling rate** je počet vzorků odebraných za sekundu při digitalizaci zvuku
- **Frame size** je čas odeslání mezi jednotlivými pakety
- **Mos** je subjektivní parametr měření. Určuje nám kvalitu hovoru. Stupnice je od jedné do pěti.
- **PSQM** – další z parametrů, které mají za úkol určovat kvalitu hovoru. Méně než pět je hovor snesitelný, méně než 4 je hovor dobrý a při hodnotě méně než tři je hovor skvělé kvality.

Nejrozšířenější je kódování **PCM** (Pulse Code Modulation). Abychom převedli signál do PCM, je zapotřebí signál vzorkovat na délky 125 mikrosekund. Poté se změří amplituda a přiřadí se jedna z diskrétních hodnot. Tato hodnota je zakódovaná do n-bitů. Většinou se kóduje do 8-bit, přičemž osmý bit nám určuje paritu.



Obr. 6: PCM

Další způsob kódování je **ADPCM** (adaptivní diferenciální PCM) – vylepšuje PCM v odhadování následujícího vzorku. Vychází z charakteristiky hlasu, který je poměrně spojitý. Také se přizpůsobuje dle druhu hlasu. Tím se snižuje dynamický rozsah a je potřebný menší rozsah na zakódování hlasu.

Kódování **LPC** je jedno z nejlepších způsobů analýzy a zpětné konstrukce hlasu pro low-bit rate kodeky. Na straně vysílací proběhne analýza hlasu, která vychází z předpokladu, že zdrojem jsou lidské hlasivky a štěrbina mezi hlasivkami produkuje bzukot, charakteristický svou hlasitostí a frekvencí. To dává hlasu jeho charakteristické zabarvení. LPC při analýze hlasu používá filtry, které odstraní zabarvení hlasu, a následně porovnává elementy zbytku hlasu s elementy ve své databázi. Při přenosu tak na stranu volaného dostaneme elementy z databáze a zabarvení našeho hlasu.

Kódování **CELP** vylepšuje LPC. LPC neumí správně aproximovat například sykot nebo různé postavení jazyka. Jednoduchý LPC systém tyto zvuky špatně rozeznává a většinou je odfiltruje. Proto CELP zavádí metodu, která tyto zvuky dokáže zachytit. Metoda spočívá ve vytvoření tabulky, kde jsou nejčastější průběhy těchto zvuků. Druhá strana má tu samou tabulku, a proto při přenosu přenáší index v tabulce a druhá strana dokáže obnovit ztracené zvuky. Mezi modifikace CELP patří **ACELP** (Algebraic Code Excited Linear Prediction), **LD-CELP** (Low Delay Code Excited Linear Prediction) a **CS-ACELP** (Conjunctive Structure Algebraic Code Excited Linear Prediction).

Následně si uděláme krátký popis nepoužívanějších kodeků:

- **G.711** je základní kodek pro komunikaci jak v IP telefonii, tak i v klasické telefonii. Bitrate má 64 kbps. Využívá kódování pomocí PCM, které je popsáno výše. Vyznačuje se dobrou kvalitou hlasu, ale poměrně velkým datovým tokem. PSQM je 4,45.
- **G.723.1** – používá se kódování buď MP-MLP nebo ACELP. V prvním případě je potřebný datový tok 6,4 a v druhém případě 5,3 bitů. PSQM je 4,08.
- **G.726** – potřebná šířka pásma je 16, 24, 32 a 40 kbps. Pracuje na principu kódování ADPCM. V podstatě nahrazuje kodek G.721. PSQM je 4,3.
- **G.729** – hodně úsporný kodek s bitrate 8 kbps. Používá pro svoje kódování CS-ACELP. Objevuje se také ve vylepšené verzi s označením G.729A a G.729B. U G.729A je vylepšená paměťová náročnost, ale trochu na úkor kvality. U verze G.729B se stejně jako u kodeku SPEEX používá technologie VAD. Často se objevuje varianta s oběma dodatky.
- **GSM** má bitrate přibližně 13 kbps a je používán jako standart pro mobilní síť. Využívá

zjednodušené kompresní metody RPE-LTP.

- **SPEEX** je svobodný kodek, který patří k projektu ogg. Dosahuje dobrých kompresních poměrů. Používá se v klasické i IP telefonii. Využívány jsou tři vzorkovací frekvence, které byly pojmenovány narrowband, wideband a ultra-wideband. Kodek SPEEX má některé zajímavé funkce – např. VAD (Voice Activity Detection), která umožňuje detekovat ticho během hovoru a tím snižovat datový tok.
- **iLBC** (internet Low Bit Rate Codec) vyvinula firma Global IP solutions. iLBC je definováno v RFC3951. Stejně jako kodek SPEEX je svobodný. Je vhodný pro VOIP aplikace, streamování audia, archivace a posílání zpráv. Používá kódování block-independent LPC. Potřebná šířka pásma je 13,3 kbps. Délka rámce je 20 a 30 milisekund.

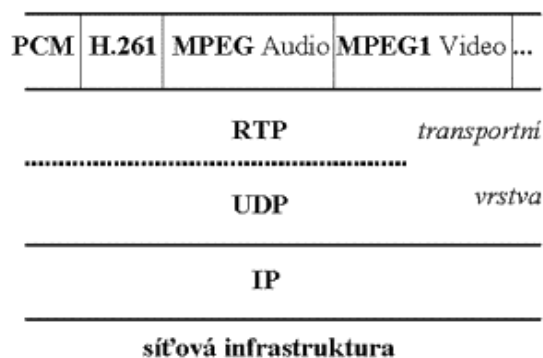
Samozřejmě, kodeků pro VOIP je mnohem více. Ty, které jsem popsal, se ale používají nejčastěji. Kdo by chtěl vědět více, najde popis některých dalších kodeků v uvedeném zdroji.

4. Protokol RTP

Zkratka RTP v angličtině znamená Real-time Transport Protocol. Anglický název přesně vystihuje tento protokol, který slouží k přenosu dat v reálném čase. Zajišťuje přenos dat pro multimediální přenosy. Funguje nad protokolem UDP. Dalo by se říci, že RTP je vlastně jakési rozšíření protokolu UDP. Využívá většinou porty 5004, 5005, 6970. Stejně jako UDP ani RTP protokol nezaručuje, že vyslané pakety k nám dorazí. U multimediálních přenosů to ani nečekáváme. Potvrzování by zbytečně zdržovalo přenos. Navíc například u hlasu už by nám znovu poslaný paket byl k ničemu, protože hovor pokračuje dál i přes ztracený paket. RTP má za úkol pakety dát do správného pořadí a vyslat k danému cíli. To má výhodu v tom, že aplikace zaznamená, když se nějaký paket zatoulá, a pomocí určitých algoritmů ho případně může dopočítat. RTP protokol byl navržen pro individuální ale i pro skupinové přenosy, a to v obou směrech. Používají ho například protokoly SIP a H.323. Ty si popíšeme v dalších kapitolách.

V současné době známe tři verze RTP protokolu:

- **RFC 1889** z roku 1996
- **RFC 3550** z roku 2003, který drobně vylepšuje předchozí verzi o dohled na RTP tokem
- **RFC 3711** – v roce 2004 vznikla zabezpečená verze SRTP (Secure Real-time Protocol)

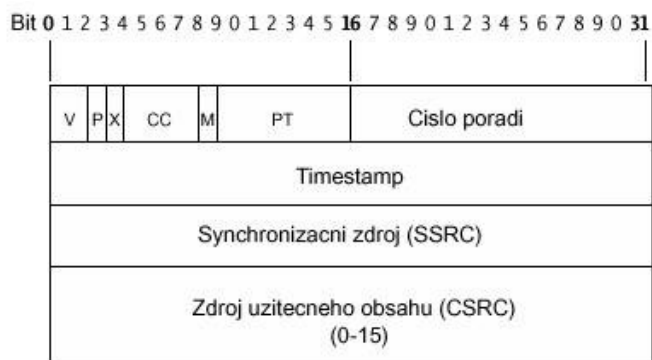


Obr. 7: RTP zařazení do síťového modelu

Jak je na obrázku vidět, RTP je řazeno nad UDP, takže by se dalo říci, že se jedná o transportní protokol – i když je řazen mezi vrstvu aplikační.

4.1 RTP RFC 1889

Na níže uvedeném obrázku vidíme, jak vypadá paket RTP ve verzi číslo 2 tedy v RFC 1889.



Obr. 8: Hlavička RTP

- Písmeno **V** znamená verzi RTP protokolu, v případě RFC 1889 je to číslo 2. (2 bity)
- **P** (padding) je dorovnání (pro šifrování). (1 bit)
- **X** je rozšiřující bit. Pokud je nastaven, pak pevné záhlaví je následováno právě jedním rozšířením záhlaví, které má definovanou pevnou délku. (1 bit)
- **CC** – content source count. (4 bity)
- **M** – interpretace této značky je různá. Použití najde např. jako ukazatel konce datového toku. (1 bit)
- **PT** (payload type) – metoda kódování audio/video, může se během relace dynamicky měnit. (7 bitů)
- **Číslo pořadí** slouží k oznámení ztráty paketu. Tato hodnota se navyšuje každým vyslaným paketem. (16 bitů)
- **Timestamp** (časová značka) – vysílač nastavuje číselnou značku paketu a přijímač použije tuto značku k rekonstrukci originálního časování pro přehrávání dat ve správném pořadí. (32 bitů)
- **SSRC** přidává ID synchronizačnímu zdroji. Dva zdroje v jedné RTP relaci nemají stejné ID. (32 bitů)
- **CSRC** je identifikační seznam přispívajících zdrojů. Identifikuje přispívající zdroj z důvodu užitečné informace obsažené v tomto paketu.

Hlavička RTP protokolu má 40 B. Každý blok dat je opatřen hlavičkou protokolu UDP, která má 8 B, následně v síťové vrstvě dostane IP hlavičku o velikosti 20 B. Hlavička RTP protokolu se dá zmenšit kompresí až na 2 – 3 B. Pak se název změní na cRTP (compressed RTP). Ovšem tuto kompresi musí podporovat obě strany, mezi nimiž probíhá datový tok. Proto se tato varianta RTP příliš nevyužívá.

Dále obsahem specifikace RFC 1889 je RTCP (Real Time Control Protocol). Jedná se o protokol pro řízení výkonnosti a diagnostiku RTP. Informuje o kvalitě poslaných dat, identifikuje zdroj RTP, synchronizuje více mediálních toků a provádí řízení intervalu vysílání RTCP.

4.2 RFC 3550

Je to novější verze protokolu RTP. Se starší specifikací je kompatibilní, protože hlavička RTP se nijak neměnila, změnilo se pouze RTCP. Nově umožňuje shromažďovat informace o multimediálním toku, jako např. počet odeslaných bajtů, počet odeslaných a ztracených paketů, Jitter (kolísání zpoždění), zpětnou vazbu a Round Trip Delay. Aplikace může tyto informace použít ke zvýšení kvality služeb, například omezením toku nebo použitím jiného kodeku. Klasické RTP dovoluje jenom jeden typ zpráv, které pouze nesou data k cíli. Ale v mnoha případech potřebujeme znát i jiné věci než jen data, která k nám přicházejí. Proto se rozšířily varianty zpráv a přidávají tak nové možnosti kontroly. Bylo definováno pět typů zpráv:

- **Sender Report** je zasílán pravidelně aktivními odesílateli v konferenci jako zprávy o přenosu a přijímací statistiky pro všechny RTP pakety poslané během intervalu. Zahrnuje v sobě také úplný timestamp, což je čas v milisekundách uplynulý od půlnoci 1. ledna 1970. Je to důležité zejména při posílání audia a videa. Používají totiž odlišná časová razítka.
- **Receiver Report** jsou zprávy, které informují příjemce i odesílatele o kvalitě služeb.
- **Source Description Message** – zdroj pravidelně posílá zprávu popisující zdroj, aby o sobě dal k dispozici více informací. Tyto informace mohou obsahovat jméno, emailovou adresu, telefonní číslo a adresu vlastníka nebo kontrolér zdroje.
- **Bye Message** ukončovací zpráva. Umožňuje zdroji oznámit, že náš stream opouští konferenci.
- **Application-Specific Message** je paket pro aplikace, které chtějí použít nové aplikace.

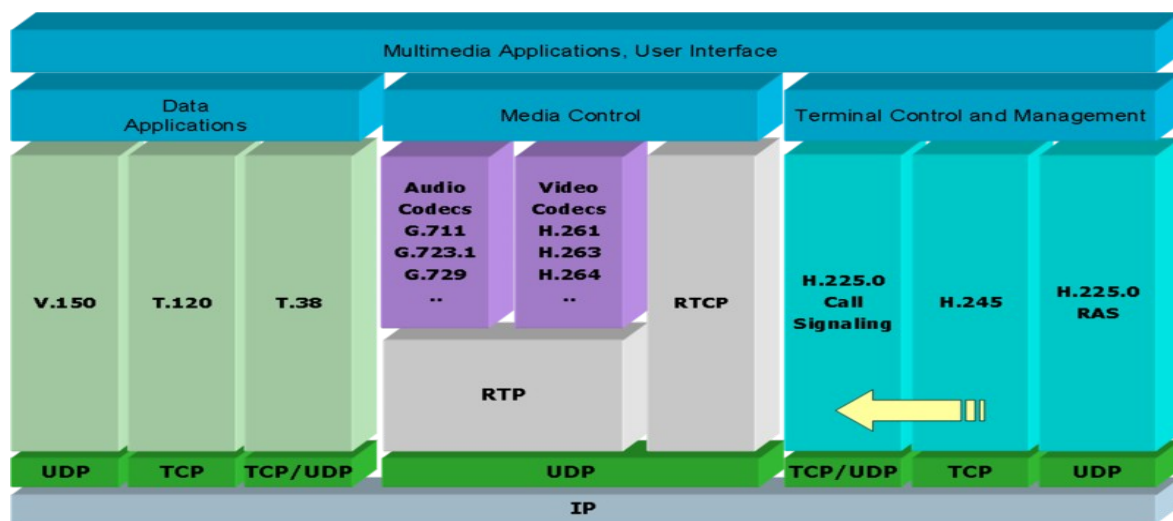
5. Signalizační protokoly

5.1 H.323

Tento standart slouží pro multimediální komunikaci v jakékoliv paketové síti. Stal se standardem vydaným pod značkou společnosti ITU-T. Poslední sedmá verze byla vydána v roce 2009. Pod tento standard spadají další protokoly :

- **H.225** – slouží k hovorové signalizaci mezi dvěma H.323 entitami za účelem navázání komunikace
- **H.245** – protokol, který má na starosti kontrolu nad přenosem multimédií. Vytváří a zavírá logické kanály pro audio, video a data.
- **H.235** – popisuje bezpečnost v H.323, ale také v signalizaci a přenosu médií.
- **H.239** – popisuje duální stream používaný při videokonferenci
- **H.450.x** – popisuje doplňkové služby jako je např. přesměrování, podržení hovoru atd.
- **H.460** – další protokol, který je určen pro videokonference, ale tento se zabývá natem a firewallem.
- **RTP** – protokol byl popsán výše.

Pro tyto protokoly se používá definiční jazyk ASN.1. Protokolové zprávy se kódují metodou ASN.1 PER (Packet Encoding Rules). Výjimku má signalizace Q.931, která používá vlastní binární kódování a také má výjimku protokol RTP, který byl popsán společností IETF.



Obr. 9: Vrstvy H.323

5.1.1 Architektura H.323

V H.323 známe čtyři entity:

Terminál – neboli endpoint (koncové zařízení) je softwarový a hardwarový telefon. Speciálním terminálem může být například i hlasový automat.

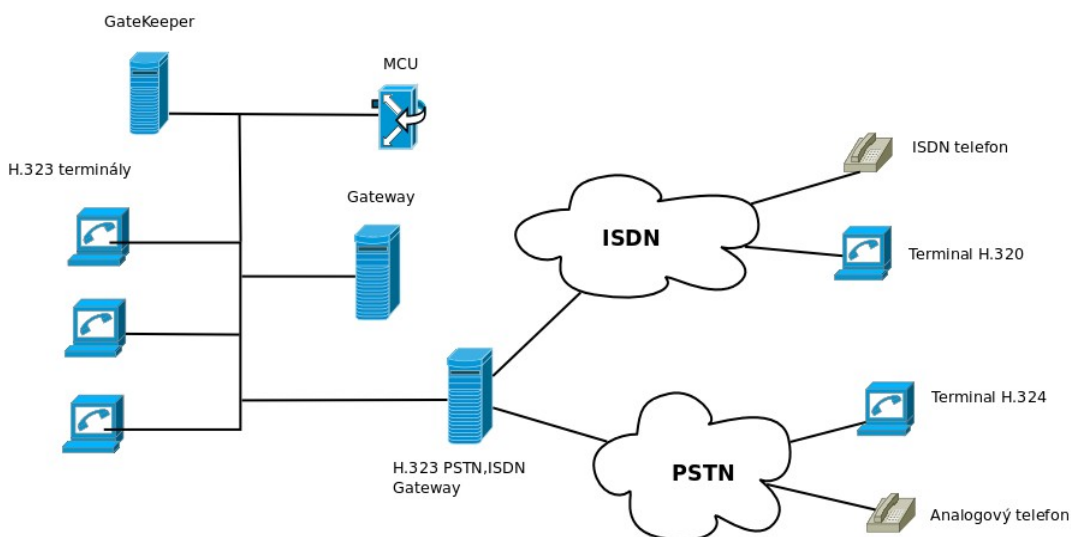
Brána – umožňuje spojení do jiných sítí – např. ISDN nebo PSTN, a také do jiné sítě H.323. Skládá se ze dvou částí: Media Gateway Controller – obsluha hovorové signalizace a Media Gateway –

směrování audio/video proudů

Multipoint control unit (MCU) – je to multikonferenční jednotka, která zajišťuje hovory mezi třemi a více body. Rozděluje se na dvě logické entity: řídicí jednotku Multipoint Controller (MC), která má na starosti obsluhu hovorové signalizace, a Multipoint procesor (jinak také Media Processor), který obsluhuje multimediální kanály, mixuje audio a přepíná video. Velkou výhodou MCU je transkódování (spojení zařízení s různými kodeky). Na obrázku č. 9 můžeme vidět vrstvy H.323. Pokud má terminál implementovány všechny prvky, které jsou znázorněny na obrázku, má veškerou multimediální komunikaci, tzn. hlas, video a přenos dat. V nejspodnější vrstvě vidíme protokol IP. Nad UDP a TCP fungují protokoly H.323. Po RTP protokolu běží hovory nebo videohovory a přes RTCP stavové a řídicí signály.

GateKeeper – volitelná součást topologie H.323. Poskytuje řadu služeb pro terminály, brány a MCU zařízení. Mezi tyto služby patří například překlad adres, konec registrace, Admission control, ověřování uživatelů atd. Asi nejdůležitější funkce je překlad adres. Umožňuje zkontaktování dvou koncových bodů aniž by museli znát svoji IP adresu. GK je řídicím prvkem H.323. Sdružuje pod sebe terminály a tvoří tak zónu. Pro každý koncový bod, který je zaregistrován na GK, musí poslouchat příkazy GK (např. příkaz ukončení hovoru). V jedné zóně nemůžeme mít více GK. Každý koncový bod dostává list s alternativními GK a tím je zajištěno, že při výpadku se zóna přidá pod jiný GK. Dalším opatřením proti výpadku je omezený čas registrace koncového bodu na GK.

Jelikož GK není povinná část topologie, všechny terminály musí dle specifikace H.323 umět komunikovat stylem point-point. Pokud chceme zajistit komunikaci mezi více terminály, musí být přítomen v zóně GK, nebo přenecháme tuto funkci MCU. MCU nám umožní multicastový přenos rozdělit na sérii unicastových přenosů a ty jsou dále rozesílány jednotlivým koncovým bodům.



Obr. 10: Entity H.323

5.1.2 Signalizace

RAS Signalizace

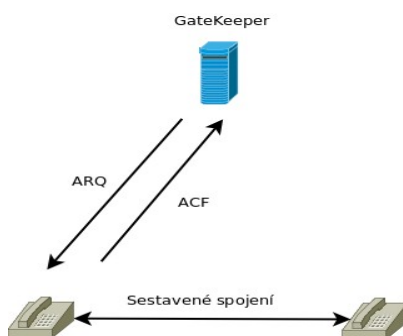
Je to poměrně jednoduchý protokol, který slouží ke komunikaci mezi GK a koncovými zařízeními.

Dále se RAS využívá ke komunikaci mezi GK. Sestává z několika zpráv:

- Registration request, reject, and confirm messages (RRx)
- Unregister request, reject, and confirm messages (URx)
- Admission request, reject, and confirm messages (ARx)
- Bandwidth request, reject, and confirm message (BRx)
- Disengage request, reject, and confirm (DRx)
- Location request, reject, and confirm messages (LRx)
- Info request, ack, nack, and response (IRx)
- Nonstandard message
- Unknown message response
- Request in progress (RIP)
- Resource availability indication and confirm (RAx)
- Service control indication and response (Scx)
- Admission confirm sequence (ACS)
- Gatekeeper request, reject, and confirm messages (Grx)

Když se zapne koncové zařízení, vyšle žádost GRQ na GK nebo zašle rovnou registrační zprávu RRQ na GK, který je předdefinovaný v nastavení. GK poté odpoví zprávou GCF (potvrzení registrace). Pokud terminál dostane toto potvrzení od GK, vyšle RRQ a GK mu pro úspěšnou registraci do zóny odpoví RCF. Poté už můžeme využívat služby zóny.

Pro inicializaci hovoru musí terminál vyslat na GK požadavek ARQ. Od GK dostane terminál v odpovědi ACF IP adresu koncového bodu, na který se chce dovolat.



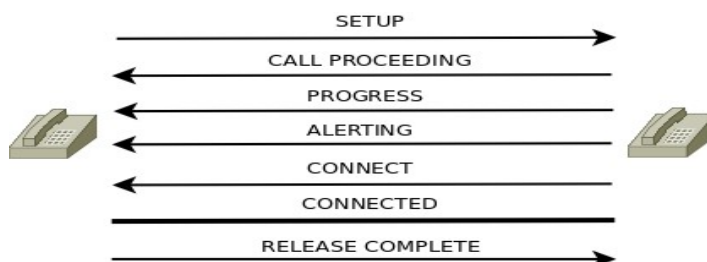
Obr. 11: Inicializace hovoru

Q.931 signalizace

Tato signalizace byla vypůjčena z L3 vrstvy ISDN. Používá se s modifikacemi v H.323 sítích a slouží k signalizaci během hovoru. Další protokol pro signalizaci je H.225, což je vlastně jinak řečený Q.931. Pro signalizaci se používají tyto zprávy:

- Setup and Setup acknowledge
- Call Proceeding
- Connect

- Alerting
- Information
- Release Complete
- Facility
- Progress
- Status a Status Inquiry
- Notify



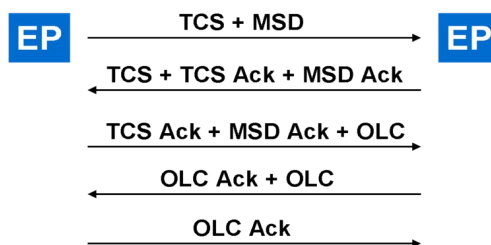
Obr. 12: Navázání hovoru

Navazování hovoru začíná zprávou Setup a ukončení hovoru se provede zprávou Release Complete. V ISDN síti se pro ukončení musí poslat dvě zprávy navíc (Disconnect, Release). Tyto zprávy se v H.323 nepoužívají kvůli tomu, že Q.931 běží nad TCP, tedy na spolehlivém protokolu.

H.245 signalizace

Je součástí Q.931 nebo se také může přenášet samostatně. Funguje na principu požadavek-odpověď. Úkolem tohoto protokolu je :

- **TCS** (Terminal Capability Set) – v tomto kroku se dohodnou kodeky, po kterých se bude komunikovat. Každé zařízení dá k dispozici list kodeků, kterými je schopno komunikovat.
- **MSD** (Master Slave Determination) – při tomto kroku se dohodne, kdo je Master, tudíž kdo má přednost v přenosu zpráv např. při nějaké chybě.
- **OLC** (Open Logical Channel) – otevře se logický kanál pro audio a video. Vybere se kodek a port. Otevře se RTP spojení zvlášť pro každý směr.



Obr. 13: Typická komunikace H.245

5.1.3 Kodeky

V H.323 je podporován jakýkoliv kodek, ať už se jedná o standardizovaný nebo proprietární. Každý terminál v tomto komunikačním protokolu by měl minimálně rozumět kodeku G.711, který

specifikovalo také ITU-T. Implementace dalších kodeků u jednotlivých terminálů závisí na výrobci. Pro video se využívají kodeky z řady H.x. Nejčastěji se používají H.264, H.263 a H.261. Pro textovou konverzaci se používá T.140, který je také definován společností ITU-T.

5.2 SIP (*Session Initiation Protocol*)

SIP je jeden ze dvou standardizovaných signalizačních protokolů, který vyvinula společnost IETF v roce 1996 jako reakci na protokol H.323. Jelikož H.323 je poměrně složitý, SIP byl vyvinut jako jeho pravý opak. První verze byla přijata v roce 1999 a nesla označení RFC 2543. V současné době existuje již druhá verze, která nahrazuje první, a nese označení RFC 3261.

Protokol slouží pro multimediální komunikaci. Umožňuje přenášet hlas, video, text, hrát síťové hry atd. SIP je textově orientovaný protokol. Má podobnou strukturu jako třeba HTTP nebo SMTP. Tělo zprávy má strukturu <název>: hodnota, která popisuje přenášené informace. Tento model se osvědčil i u zmiňovaného HTTP.

Protokol je typ klient – server. To znamená, že probíhá výměna dvou zpráv – požadavků a odpovědí. Klient zasílá dotazy a server na ně odpovídá. V hlavičce nalezneme From, To a Subject, jako je tomu u SMTP protokolu. SIP patří mezi protokoly aplikační vrstvy a pracuje většinou nad UDP na portu 5060, ale možné je i použití TCP a portu 5060.

5.2.1 Adresace v SIP

Jednotliví uživatelé jsou identifikováni adresou SIP neboli také SIP uri (Uniform resource identifikátor). Většinou se v praxi setkáme s jednoduchým obecným tvarem: sip:user@host. Za hostem je možné ještě uvést další nepovinné parametry jako např. číslo portu, pokud je odlišné od běžného portu 5060.

Jak je vidět, adresa se značně podobá mailové adrese. Tedy mimo předponu sip:. Ta nám rozlišuje, o jaký protokol se jedná. User může být jméno, ale také číslo. Za zavináčem se nachází host, který může být napsán jako IP adresa nebo jako doménové jméno, např. hlas.802.cz.

5.2.2 Architektura SIP

Jak jsem již zmínil, SIP pracuje na principu klient-server. V nejjednodušší variantě topologie můžou být jen dvě koncová zařízení. Ale většinou se setkáme s variantou, kdy v síti budeme mít více prvků. Sip topologie se skládá z těchto částí:

- koncový bod UA (User Agent)
- Proxy, redirect, registrar server

Proxy, redirect a registrar jsou logické celky SIP serveru, které se většinou provozují na společném hardwaru. Ale samozřejmě mohou být i oddělené. Koncovým bodem neboli user agents, je myšlen hardwarový nebo softwarový telefon, mobilní telefon, pda, ivr systémy, brány atd. UA má dvě logické části, UAS (User Agent Server) a UAC (User Agent Client). UAS je část, která vysílá odpovědi a přijímá požadavky. UAC naopak vysílá požadavky a přijímá odpovědi. Tyto dvě role zůstávají jen po dobu transakce. Poté se role může vyměnit. To znamená, že ve chvíli, kdy volající pošle INVITE, stává se UAC a volaný je UAS. Vyměnit se mohou ve chvíli, kdy volaný ukončí hovor zprávou BYE. V tuto chvíli se stává volaný UAC a volající UAS.



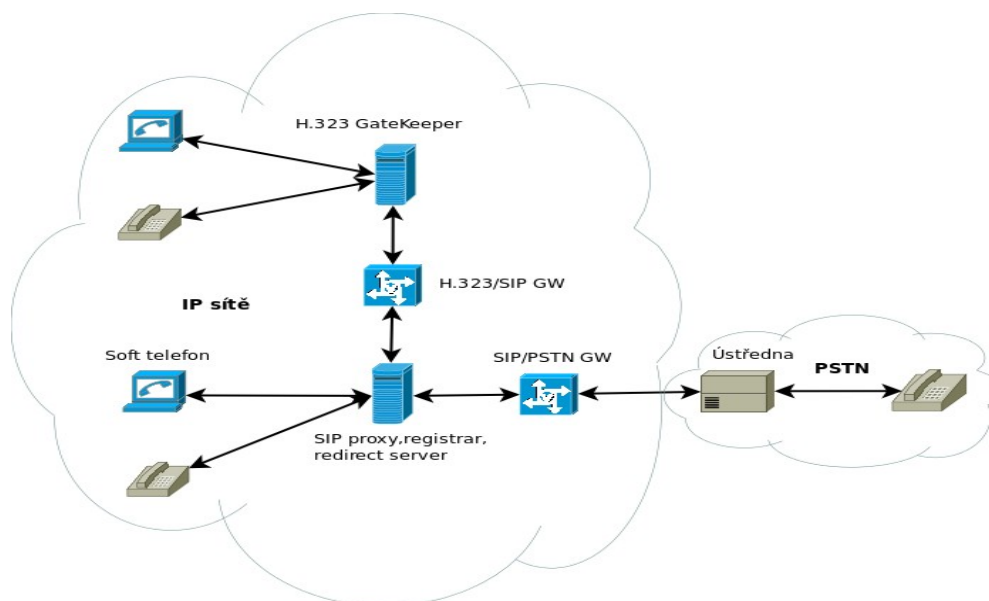
Obr. 14: Role UAC a UAS

SIP server je obvykle zodpovědný za uživatele v doméně. To znamená, že SIP je decentralizovaný a vytváří infrastrukturu hostitelů – **SIP proxy serverů**. Tyto servery zajišťují směrování žádostí o spojení dle lokace uživatele, mohou provádět registraci, autentizaci a jiné doplňkové služby. Nejdůležitější funkcí je směrování. Proxy server např. při inicializaci hovoru bude hledat u ostatních proxy serverů, zda nezná umístění koncového zařízení. Umístěním se myslí IP adresa, uživatel a port. Po nalezení umístění už proxy server dokáže nasměrovat požadavek INVITE k volanému. Rozeznáváme dva druhy proxy serveru:

- **Stateless** (bezstavový) – jsou to jednoduché servery, ale mnohem rychlejší než Stateful servery. Umí jen přeposílat zprávy, ale nemají přehled o stavu hovoru.
- **Stateful** – jsou rozmanitější. Umí si držet informace o stavu hovoru. Negativem je, že zatěžují server. Proto je výkon u těchto serverů limitován. Ale na druhou stranu získáme zajímavé funkce, jako je třeba přesměrování hovorů na více zařízení (pokud se nedovoláme na jedno zařízení, provede se přesměrování na další). V praxi se setkáme převážně s tímto druhem proxy serveru. Dělí se na dvě skupiny:
 - transakční – drží si stav žádosti, dokud není vyřízena
 - dialogové – drží stav, dokud neskončí celé spojení

Registrar server je většinou logickou částí proxy serveru. Jedná se o speciální případ UAS. Má za úkol přijímat požadavky registrace od klientů. Tím získá jejich umístění (IP adresu, port, jméno) a následně toto umístění uloží do databáze location service. V lokalizační databázi je poté uloženo třeba URI abcd@matty-net.cz a na to je namapováno device URI abcd@1.2.3.4:5060. Až bude chtít proxy server vyhledávat abcd@matty-net.cz, dostane adresu abcd@1.2.3.4:5060.

SIP Uri je vázáno k doméně. Proto když například změním svoji doménu, je zapotřebí dát vědět, že se nám lidé mají dovolat do jiné domény. A k tomu právě slouží **Redirect server**. Redirect server umí přesměrovat naše volání na správnou doménu. Pokud budu mít URI abcd@matty-net.cz, změním ho na abcd@test.cz. Někdo mi bude chtít zavolat, vyšle zprávu INVITE na abcd@matty-net.cz. Ale jelikož mám už jinou doménu, vrátí mi Redirect server odpověď 302 Moved Temporarily a v této odpovědi bude kolonka contact, kde bude zapsáno moje nové URI. A na toto nové Uri se už dovoláme.



Obr. 15: Topologie sítě

5.2.3 Požadavky

Jednotlivé požadavky jsou posílány textově v datagramech UDP nebo TCP. Žádosti slouží například k požadavky o sestavení, k registraci na SIP server nebo i k ukončení hovoru.

Příklad požadavku INVITE:

```
INVITE sip:mamut@iptel.org SIP/2.0.
Max-Forwards: 10.
Record-Route: <sip:195.113.222.3;ftag=5DAA94E7;lr=on>.
Via: SIP/2.0/UDP 195.113.222.3;branch=z9hG4bK0a5d.90580ee2.0.
Via: SIP/2.0/UDP 195.113.134.233:5062;branch=z9hG4bK2E1FD348.
CSeq: 262 INVITE.
To: <sip:mamut@iptel.org>.
From: "Franta Vomacka" <sip:bbb@ces.net>;tag=5DAA94E7.
Call-ID: 379332994@195.113.134.233.
Subject: sip:bbb@ces.net.
Content-Length: 234.
User-Agent: kphone/4.2.
Contact: "Franta Vomacka" <sip:bbb@195.113.134.233:5062;transport=udp>.
.
v=0.
o=username 0 0 IN IP4 195.113.134.233.
s=The Funky Flow.
c=IN IP4 195.113.134.233.
t=0 0.
m=audio 33728 RTP/AVP 0 97 8 3.
a=rtpmap:0 PCMU/8000.
a=rtpmap:3 GSM/8000.
a=rtpmap:8 PCMA/8000.
a=rtpmap:97 iLBC/8000.
a=fmtp:97 mode=30.
```

Obr. 16: Žádost INVITE

- V prvním řádku zprávy se dozvíme, že se jedná o žádost INVITE. Dále v tomto řádku nalezneme informaci Request Uri (sip:mamut@iptel.cz), které je dalším skokem (next hope) zprávy.
- **Via** – jsou cesty, přes které putuje žádost, a přes které se bude vracet také odpověď
- **Cseq** – pořadové číslo žádosti v dialogu (o dialogu si povíme dále v textu). Každá nová žádost inkrementuje Cseq. Ale zaslání odpovědi na žádost se neinkrementuje.

- **To** – identifikuje volaného neboli příjemce volání.
- **From** – indetifikuje volajícího neboli iniciátora volání.
- **Max-forwards** – maximální počet proxy serverů na trase.
- **Call id** – identifikátor dialogu. Pomocí něho můžeme identifikovat zprávy náležící jednomu volání.
- **Contact** – obsahuje port a IP adresu, na kterých odesílatel čeká odpověď. Jelikož si obě strany vymění tento kontakt, nemusí se využívat sip proxy, když chce jedna ze stran ukončit hovor.
- **Record-Route** – seznam adres serverů, které chtějí dostat zprávy o probíhajícím hovoru.

Jak je na obr. 16 vidět, požadavek obsahuje také prázdný řádek, který odděluje hlavičku zprávy od těla. V těle zprávy se dohodnou kodeky pro hlas i video popsané protokolem SDP.

Druhů zpráv pro požadavky je několik:

- **Invite** – tato zpráva slouží k inicializaci hovoru nebo také pro změnu parametrů hovoru.
- **Bye** – ukončení hovoru.
- **ACK** – je potvrzením pro INVITE. Když volaný přijme INVITE, pošle potvrzující zprávu 200 OK a volající pošle zprávu ACK.
- **CANCEL** – volaný touto zprávou odmítne hovor.
- **REGISTER** – pro registraci nebo odregistrování na SIP server je nutné zaslat tuto zprávu. Registrace má omezenou dobu a je nutné ji obnovovat.
- **OPTIONS** – používá se k zjištění informací o zařízení.
- **INFO** – přenos signalizačních zpráv.
- **MESSAGE** – přenos zpráv pro Instant Messaging.
- **NOTIFY** – doručení o události.
- **PUBLISH** – aktualizace stavu klienta.
- **SUBSCRIBE** – přihlášení k upozornění na nějakou událost.
- **PRACK** – potvrzení dočasné odpovědi (1xx).

5.2.4 Odpovědi

Je-li poslána žádost, musí na ni být dána i odpověď. Odpověď vypadá podobně jako žádost, pouze s tím rozdílem, že v prvním řádku se posílá verze protokolu SIP, což je v současné době SIP/2.0 a také se posílá číselná hodnota v rozmezí 100 – 699. Aby bylo jasné o jakou odpověď se jedná, je k ní vždy připojen text, např. OK (číselná hodnota 200). Existuje šest tříd odpovědí:

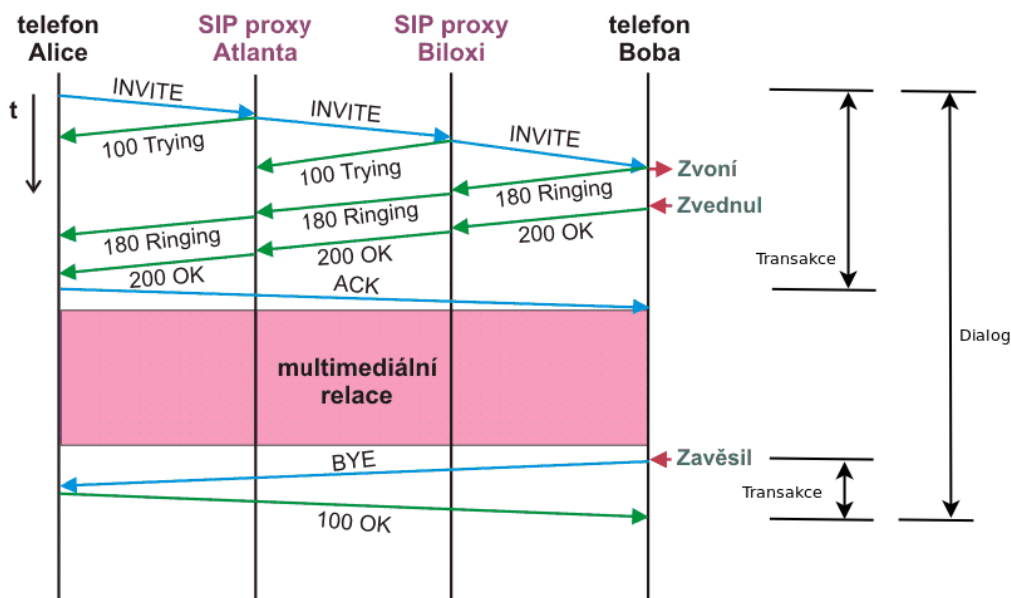
- **1xx** – Zpráva o přijetí požadavku, ale ještě není znám výsledek. Tato odpověď zamezí posílání dalších požadavků.
- **2xx** – Potvrzení o úspěšnosti požadavku. Tyto odpovědi jsou konečné a ukončují transakci.
- **3xx** – Tyto odpovědi obsahují informace o přesměrování. Pokud volaný má nové URI a volající volá na staré URI, dostane tuto odpověď a v ní novou SIP adresu, kam se má hovor přesměrovat.
- **4xx** – Jsou to negativní zprávy. Informují o chybě na straně klienta. Většinou se jedná o

špatnou syntaxi požadavku, který nemohl být zpracován.

- **5xx** – Také negativní zpráva, ale tentokrát ze strany serveru.
- **6xx** – Globální zpráva, že požadavek nedokáže zpracovat žádný server.

5.2.5 Transakce a dialogy

Transakce je soubor zpráv, které si posílají SIP entity mezi sebou. Začíná požadavkem a končí jednou nebo vícero konečnými odpověďmi. Mezi požadavkem a konečnou odpovědí může být žádná nebo několik dočasných zpráv. I přes to, že se posílají zprávy nezávisle na sobě, UA je uspořádává do transakcí.

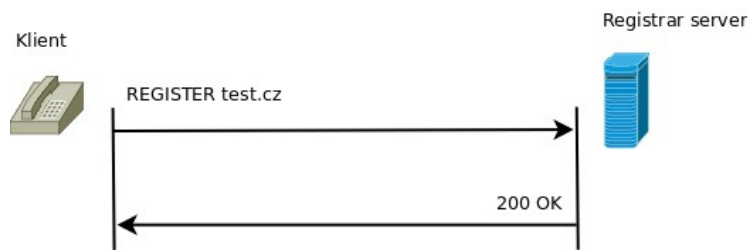


Obr. 17: Transakce a dialog

Dialog je sestaven z transakcí. Dialog by se dal nazvat jako jeden hovor. Jak vidíme na obrázku č.17, ve znázorněném dialogu proběhly dvě transakce – INVITE a BYE. Zprávy z jednoho dialogu se dají poznat dle tří identifikátorů (From, To, Call-ID). U těchto identifikátorů se nachází tag, který když je stejný, pak zprávy náleží do téhož dialogu.

Na obrázku č. 17 vidíme nejen transakce a dialog, ale také průběh navazování a průběh a ukončení hovoru. Když chce Alice zavolat Bobovi, pošle zprávu INVITE. Než dorazí k Bobovi žádost o spojení, dostává Alice zprávu 100 Trying. To znamená, že její požadavek je správný, ale ještě není znám výsledek. Poté co dorazí k Bobovi INVITE, je zpět poslána zpráva 180 Ringing. Alice tedy uslyší ve svém přístroji zvonění. Bob zvedne telefon, k Alici dorazí zpráva 200 OK a Alice pošle potvrzení ACK. A hovor je navázán. Tím nám skončila jedna transakce.

Dalším příkladem transakce je registrace, která nastane, pokud klient odešle na registrar server správné údaje. Registrace sestává z požadavku REGISTER a odpovědi 200 OK. Pokud klient pošle údaje nesprávné, zpět putuje odpověď 401 Unauthorized.



Obr. 18: Registrace klienta na server

Praktická část

6. Realizace IP ústředny

K realizaci IP ústředny budeme potřebovat technické vybavení a také potřebný software. Ze softwaru se jako nejvíce vhodný jevil **Elastix**. Je to linuxová distribuce založená na CentOS (současná verze 5.4 final). Tato IP PBX ústředna je založená na projektu Asterisk a FreePBX, které je webovým rozhraním nad Asteriskem. Elastix toho nabízí ovšem mnohem víc. Umožňuje spravovat nejen volání, ale umí přes různé moduly používat také e-mail či Instant Messaging, které je založeno na protokolu XMPP. Proto instalaci této IP ústředny máme v ruce velice mocný nástroj na vybudování serveru, který nám umožní komunikovat nejen hlasem. Navíc vše je programováno pro co nejjednodušší správu. Elastix tak zvládne naprogramovat i člověk, který neví jak správně nakonfigurovat Asterisk. A co k této realizaci budeme potřebovat?

- Stáhnout obraz ze stránek www.elastix.org v sekci download. Já jsem zvolil verzi beta **Elastix 2.0 RC1**.
- **ISDN BRI PCI kartu openVox B100P 1-Port** – umožňuje režim NT/TE a zvládá 2 samostatné hovorové kanály
- **Počítač** – konfigurace je závislá na tom, kolik lidí na ústřednu chceme připojit. Pro testovací účely by mělo stačit PC s Intel Celeron/Pentium 3 500 MHz, 512 Mb ram a 20 Gb disk. Ovšem pro nasazení, kdy bude připojeno 200 lidí a bude probíhat 50 souběžných hovorů se doporučuje Intel Core 2 a 4 Gb paměti.
- Pro napojení na VOIP providera bychom měli mít zřízený účet. A také bychom měli mít zjištěny informace od našeho poskytovatele pro konfiguraci. Upozorňuji, že konfigurace se dle providera může lišit. V tomto návodu je napojení Trunku na **poskytovatele 802**.

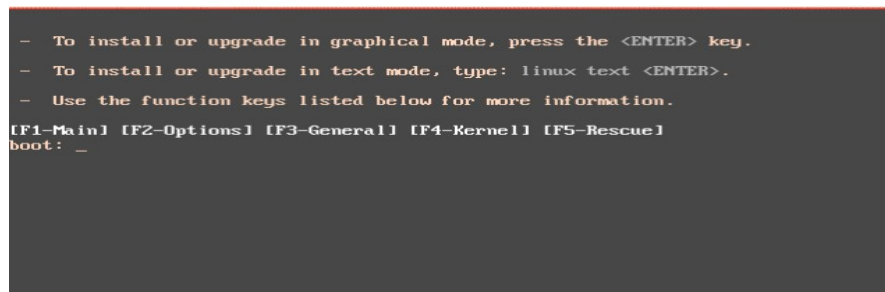
Pokud máme vše, co je uvedeno výše, jsme schopni zrealizovat IP ústřednu, která je napojena na ISDN linku a pomocí SIP Trunku se připojíme na poskytovatele VOIP telefonie. Toto napojení má výhodu v tom, že pokud nám vypadne např. ISDN linka, můžeme volat přes VOIP providera. Ovšem když nám nepojede internet ani ISDN, je možné volat si jen v rámci vnitřní sítě.

6.1 Instalace

Po stažení ISO obrazu jej vypálíme na CD nebo DVD. Předpokladem pro další krok je nainstalovaná PCI ISDN karta v počítači. Aby se nám spustilo boot CD, musí být v biosu povoleno bootování z CD. Po vložení CD by se mělo zobrazit to, co je znázorněno na obrázku č. 19. Instalátor nabídne dvě možnosti:

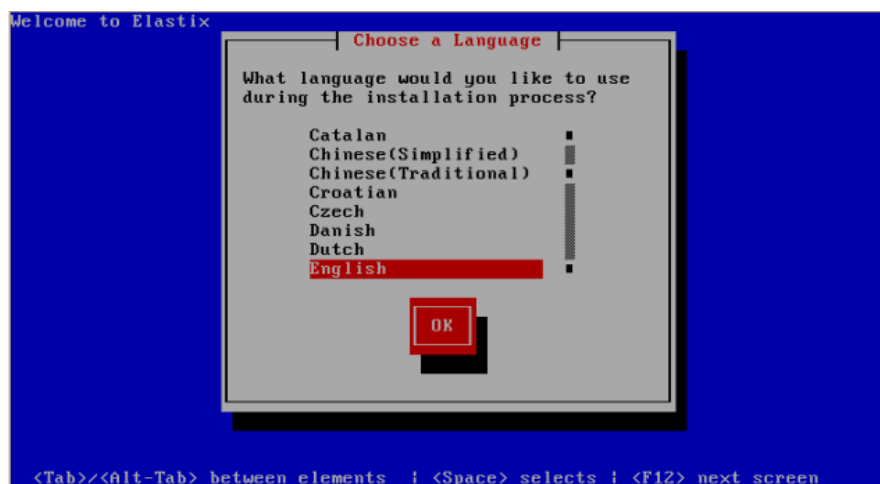
- Instalaci v grafickém módu, což je doporučená instalace
- Instalaci v textovém módu, kdy instalace probíhá na základě dotazu.

Pokud bychom chtěli zvolit některé speciální vlastnosti instalace, dostaneme se k nim přes zkratky, které lze vidět na obrázku. Pro náš účel bude ale postačující mít vše v přednastavených hodnotách. Proto pro instalaci Elastixu stiskneme **Enter**.



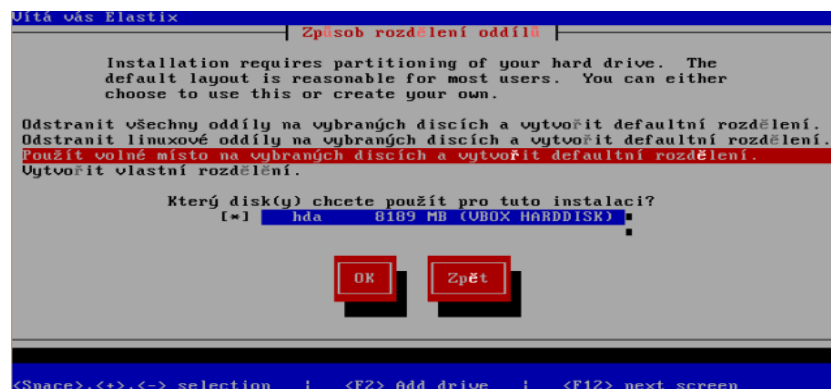
Obr. 19: Úvodní obrazovka instalace

Dalším krokem bude volba jazyka. Elastix podporuje český jazyk, tedy můžeme zvolit češtinu. Ovládání je popsáno níže na obr. č. 20. Pro potvrzení zmáčkneme klávesu **Tab** a poté **Enter**.



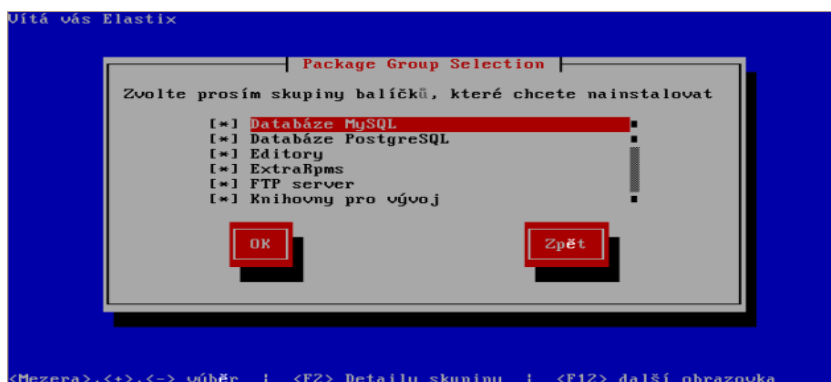
Obr. 20: Výběr jazyka

Následuje výběr libovolné klávesnice. Důležitým krokem je pak rozdělení disku. Pokud máme zcela nový disk bez vytvoření tabulky oddílů, systém se zeptá, zda chceme jej zformátovat. Následujícím krokem je rozdělení disku. Jestliže člověk s Linuxem nemá příliš velké zkušenosti, doporučuji využít některou z možností defaultního rozdělení disků. Jelikož jsem měl vyhrazený disk pouze pro Elastix, zvolil jsem první možnost. Tedy odstranit vše na disku.



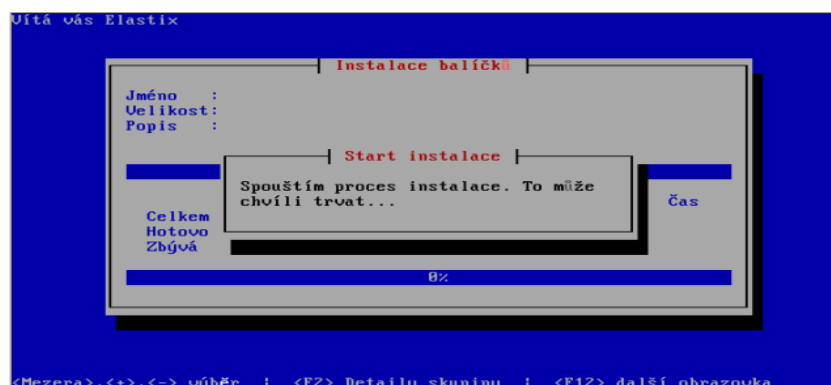
Obr. 21: Rozdělení disku

Vybereme časové pásmo a objeví se obrazovka volby hesla pro uživatele **root**. Heslo volíme bezpečné, nejlépe kombinaci písmen a čísel. Uživatel root má totiž neomezená práva v systému, proto je důležité mít bezpečné heslo. Poslední krok před konečnou instalací je výběr balíčků pro sestavení systému. Pokročilý uživatel Linuxu už je schopen rozeznat, zda ten který balíček bude potřebovat, či nikoliv. Jestliže patříme k začátečníkům, je doporučeno nechat toto nastavení ve výchozích hodnotách.



Obr. 22: Výběr balíčků

Jestliže se nám zobrazí zpráva, která je vidět na obrázku č. 23, byli jsme úspěšní a systém se nám instaluje do počítače.



Obr. 23: Začátek instalace

6.2 Konfigurace Elastixu

Pokud jsme ve fázi, kterou znázorňuje obrázek č. 24, instalace zdárně proběhla a nyní nás čeká konfigurace Elastixu. Nastavování je celkem pohodlné, protože se tak děje přes velice přívětivé webové rozhraní. Tím je pro uživatele konfigurace velmi zjednodušena. Bez tohoto ulehčení bychom museli psát do konfiguračních souborů, ale takto systém udělá vše za nás. Ale ještě než přistoupíme k webovému rozhraní, doporučuji updatovat systém. K tomu je ovšem zapotřebí internet. Jestliže nemáme přístup na internet z důvodu špatného nastavení, je možné toto napravit přes příkazovou řádku.

Do příkazové řádky se dostaneme zadáním loginu root a hesla, které jsme zvolili při instalaci. Když chceme nakonfigurovat síť, třeba z důvodu nastavení pevné IP adresy, zadáme např. příkaz **ifconfig <název síťové karty (např. eth0)> 192.168.1.1 netmask 255.255.255.0**. Zkontrolování sítě provedeme příkazem **ifconfig <název síťové karty (např. eth0)>**. V defaultním nastavení je nastavena síť pro automatické získávání IP adresy.

Když nejsme zdatní v linuxu. Nabízí se možnost nastavit síť přes webové rozhraní. Ovšem je nutné nastavit minimální konfiguraci v předchozím kroku. Obrázek níže ukazuje, jak vypadá prostředí pro nastavování sítě. Vidíme tu naše síťové karty a jejich parametry pro síť.

The screenshot shows the 'Network Parameters' section of the Elastix web interface. It includes a form for editing network settings. Below this, the 'Ethernet Interfaces List' is displayed as a table with columns for Device, Type, IP, Mask, MAC Address, HW Info, and Status. One interface, 'Ethernet 0', is listed with a DHCP type and a 'Connected' status.

Device	Type	IP	Mask	MAC Address	HW Info	Status
Ethernet 0	DHCP	192.168.56.101	255.255.255.0	08:00:27:39:D8:8F		Connected

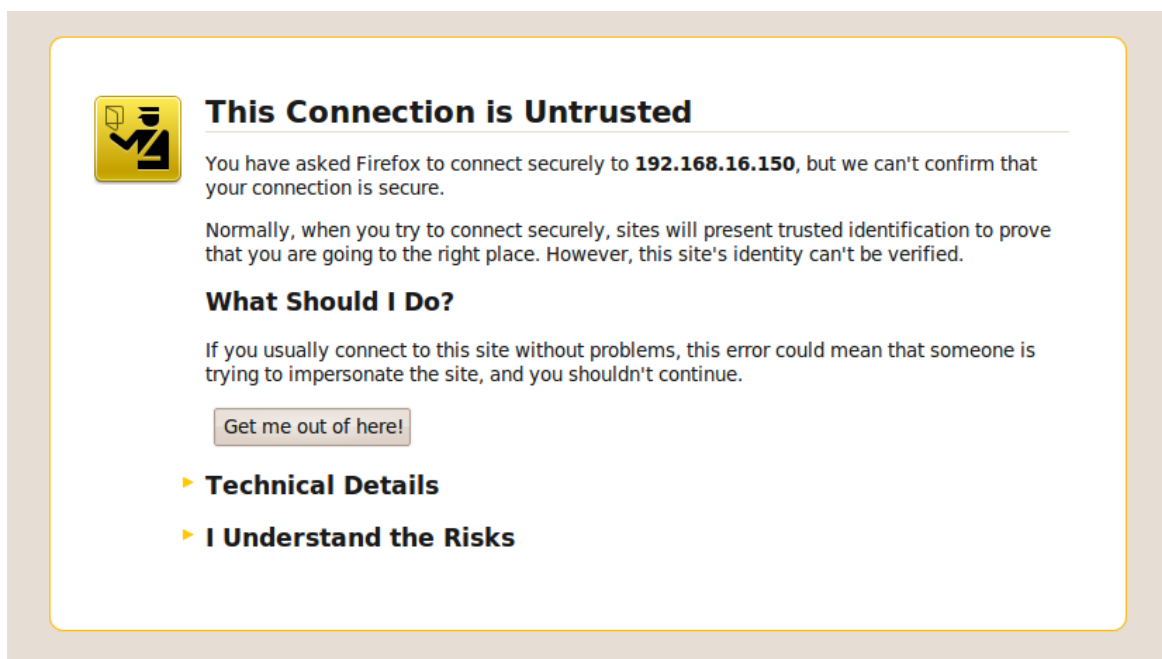
Obr. 24: Konfigurace sítě



Obr. 25: Úvodní obrazovka do systému CentOS

Pro pohodlnější správu lze využít dálkovou správu SSH, abychom nemuseli fyzicky být u serveru. Přihlášení do systému dálkově provedeme buď přes příkazovou řádku v Linuxu zadáním **ssh root@<IP adresa serveru>** nebo ve Windowsu použitím programu Putty.

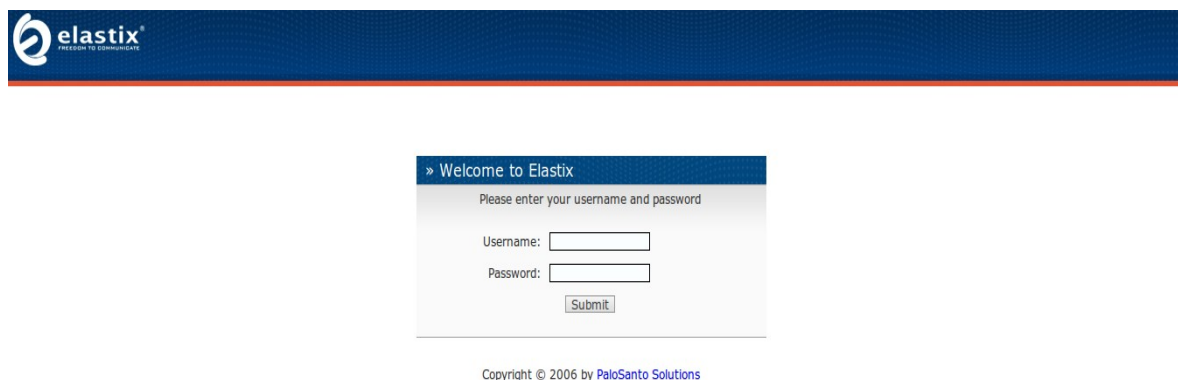
A nyní se podíváme do webového rozhraní. Spustíme tedy prohlížeč a zadáme IP adresu našeho serveru. Jak vidíme na obrázku č. 26, dostaneme hlášku o zabezpečení. Odklikneme, že jsme srozuměni s bezpečnostním rizikem a přidáme výjimku pro tento server.



Obr. 26: Chyba certifikátu

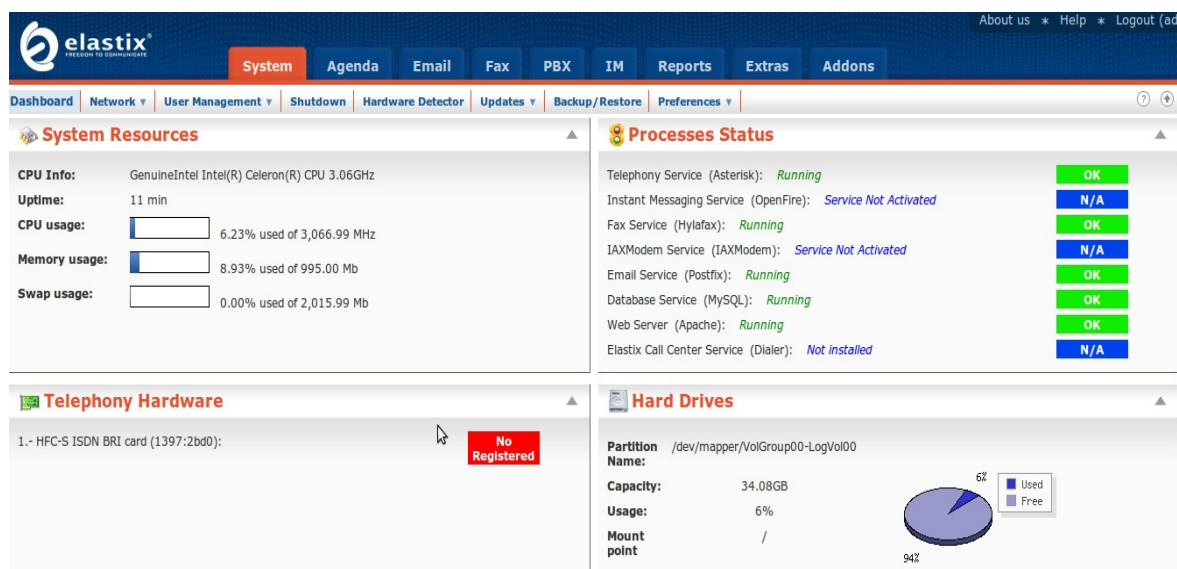
Jestliže jsme výjimku přidali, vítá nás systém Elastix svojí úvodní obrazovkou a čeká na zadání přihlašovacích údajů. Ve výchozích hodnotách je uživatelské jméno **admin** a heslo je **palosanto**.

6.2.1 Informační tabule a změna hesla admin účtu



Obr. 27: Uvítací obrazovka webového rozhraní Elastixu

Po přihlášení se otevře prostředí **Elastixu**. Je členěno do přehledných záložek a v každé záložce najdeme další nástroje pro správu. Nyní jsme v záložce, která obsahuje celkové nastavení systému a prostředí. Na obrázku č. 27 je Dashboard čili informační deska, která ukazuje celkové informace o systému, jako je přehled o spuštěných službách, o detekovaném hardwaru, o zaplnění místa na disku atd. Tyto informační widgety si můžete dle libosti vypínat a zapínat. Cesta ke správě apletů je **Preferences → Aplet Admin**.



Obr. 28: Informační tabule po přihlášení

Doporučuji ihned změnit heslo k administrátorskému účtu, aby se zamezilo vniknutí běžných i nechtěných uživatelů. To lze snadno provést v sekci **System** → **User management** → **Users**. V User Management můžeme přidávat uživatele, přiřazovat je do skupin a skupinám udělovat práva do systému. Pro změnu účtu administrátora přejdeme do sekce **Users** a klikneme na uživatele **admin**. Administrátorovi můžeme přiřadit telefonní číslo, heslo a jméno, jak je vidět na obr. 28.

The screenshot shows the 'Edit User "admin"' form in the Elastix web interface. The 'User Management' tab is active, and the 'Users' sub-tab is selected. The form contains the following fields:

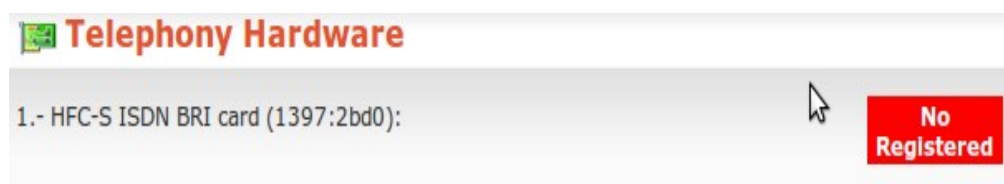
- Login:** admin
- Password:** (empty field)
- Group:** administrator
- Name (Ex. John Doe):** (empty field)
- Retype password:** (empty field)
- Extension:** (dropdown menu)

Buttons for 'Apply changes' and 'Cancel' are visible. A red asterisk indicates required fields.

Obr. 29: Změna informací účtu administrátora

6.2.2 Instalace karty

Při spuštění webového rozhraní jsme si mohli povšimnout apletu detekce hardwaru (obr. 30). Zatím je ISDN karta neregistrovaná. Jednoduchým krokem ji zprovozníme.



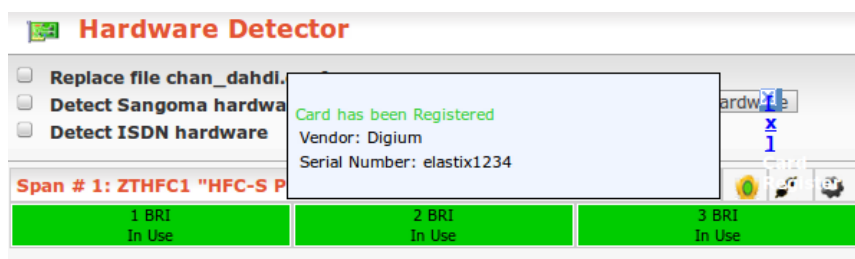
Obr. 30: Neregistrovaná ISDN karta

Jdeme do sekce **System** → **Hardware Detector**. Vidíme, že naše karta byla detekována. V případě, že náš hardware nebyl nalezen, můžeme označit volby pro nalezení a stisknout tlačítko **Detect New Hardware**. Jestliže ani po tomto kroku karta nebyla nalezena, musíme hledat informace u našeho výrobce ISDN karty.



Obr. 31: Detekce hardwaru

Na obr. 31 jsou tři neznámé kanály karty. Pro aktivaci kanálů zmáčkne nejbližší tlačítko k nápisu master. Všechna tři políčka by se měla rozsvítit zeleně, tak jak je tomu na obr. 32. Objeví se pole s požadavkem na registrační údaje karty. Ty je potřeba zadat v informační tabuli jak je tomu na obr. 33. Pod zbývajících tlačítky najdeme volby např. pro potlačení echa, signalizaci, použité kódování atd. Pro naše účely postačí, když ponecháme výchozí hodnoty.



Obr. 32: Aktivace kanálů

Registrační údaje si zaznamenáme a okno s registračními údaji zavřeme kliknutím na modrý křížek. Následuje přesun zpět do sekce **System** → **Dashboard**. V apletu **Telephony hardware** zmáčkne červené tlačítko **No Registered**. V okně, které se objeví, vyplníme údaje získané při předchozím kroku.



Obr. 33: Registrace karty

Pokud na nás svítí z monitoru u ISDN karty zelené políčko **Registered**, zvládli jsme instalaci karty.

6.2.3 Extensions (Volající uživatelé)

Velice důležitou funkcí je nastavení ústředny v sekci PBX. První, co nás v této záložce čeká, je vytvoření volajících uživatelů. Vybereme možnost vytvoření **Generic SIP Device** (obr. 34) a stiskneme tlačítko **Submit**. Měla by se nám zobrazit stránka s volbami k vytvoření účtu (obr. 35).



Obr. 34: Přidání SIP účtu

Pro zřízení fungujícího účtu stačí vyplnit tři políčka:

- **User Extension** – číslo, na které budeme volat. Slouží také jako uživatelské jméno pro registraci na náš server.
- **Display Name** – identifikace volání (jméno volajícího).
- **Secret** – heslo, které bude zadávat uživatel při registraci k PBX.

Ostatní volby jsou pro naše konfigurování nepotřebné. Pokud by se někdo chtěl dozvědět další možnosti zřízení účtu, může se podívat do oficiální dokumentace Elastixu. V případě, že máme vyplněny potřebné údaje, stiskneme tlačítko **Submit**. Poté je nutné restartovat ústřednu – tato možnost volby by se nám měla zobrazit nahoře pod záložkami po stlačení Submitu.



Obr. 35: Tlačítko pro restart ústředny a aplikování změn

6.2.4 SIP a ISDN Trunk

Trunk je napojení na různé sítě jako je GSM, ISDN atd. V podstatě naše privátní ústředna je připojena k ústředně poskytovatele. Trunk lze také chápat jako klienta, který se připojuje na ústřednu. Přes toto napojení lze uskutečňovat a přijímat hovory. Např. když budu chtít volat do ISDN sítě a ne do místní, zmáčknou volbu např. 0, která bude směřovat hovory na ISDN linku. Ale nyní se budeme věnovat konfiguraci napojení na SIP providera a ISDN.

Na obr. 36 je v levém sloupci označena volba **Trunk**. K této volbě se dostaneme kdekoli ze systému otevřením záložky **PBX → PBX Configuration**. Klikneme tedy na volbu Trunk a před sebou vidíme výběr různých napojení, které můžeme zřídit s naší ústřednou. Nás zajímá Trunk Zap čili napojení na ISDN. Ve výchozím stavu takový Trunk už je vytvořený, nachází se vpravo a jmenuje se Zap channel g0. Tudíž nemusíme nic konfigurovat. Ale přesto si ukážeme, jak na to.



Obr. 36: Výběr Trunku

Zvolíme **Add Zap Trunk**. V případě konfigurace ISDN Trunku nás budou zajímat dvě kolonky, které jsou označeny červeným rámečkem na obr. 37.

Obr. 37: Vytvoření Trunku

- **Description** – název Trunku. Můžeme zvolit jakékoliv jméno.
- **Zap Identifier** – zde už musíme vyplnit hodnotu g0, což znamená, že ISDN karta byla přiřazena do group =0. Toto za nás udělala detekce hardwaru.

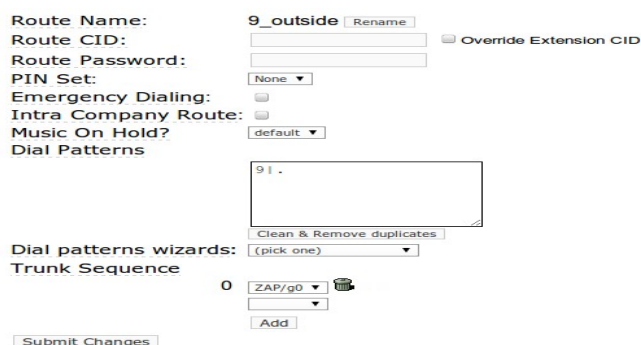
Opět klikneme na tlačítko Submit a aplikujeme změny. Tím je náš Trunk připravený. Ovšem to není všechno. V tuto chvíli by se nám nikdo nedovolal a ani my bychom si nezavolali. Abychom tuto skutečnost napravili, musíme definovat pravidla pro odchozí a příchozí hovory.

Odchozí hovory se konfiguruji v sekci **PBX → PBX Configuration**, kde v levém panelu by měla být volba **Outbound routes**. Implicitně je vytvořené pravidlo pro odchozí volání přes volbu 9. Klidně si můžeme vytvořit jakoukoliv volbu, přes kterou budeme volat. Při zakládání nového pravidla pro odchozí volání nás budou zajímat tři věci:

- **Route name** – jméno odchozího směru
- **Dial Patterns** – neboli dial plán. Pomocí pravidel můžeme určit, za jakých podmínek bude puštěno vytáčené číslo do Trunku.
 - | je oddělovací znak volby od vytáčeného čísla např. 9|.. Volba 9 nejde do Trunku.
 - X je jakékoliv číslo z rozsahu 0-9.
 - Z je číslo z rozsahu 1-9.
 - N je číslo z rozsahu 2-9.
 - . je zástupcem pro jakýkoliv znak.
 - + znaky před tímto symbolem budou přidány k vytáčenému číslu. Např. 420+ 123456789 (do Trunku půjde 420123456789).
 - [] zastupuje jeden znak. V závorkách jsou znaky, které jsou povoleny, např. [1-3].

Můžeme použít pouze čísla 1, 2, 3, žádný jiný znak není povolen.

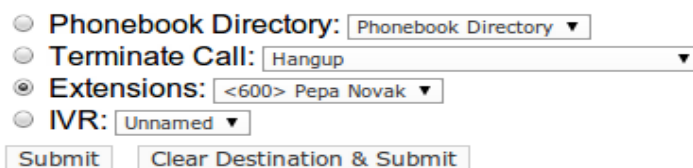
- **Trunk sequence** – Trunk, pro který bude platit odchozí směr



Obr. 38: Přidání odchozího volání pro Trunk

Opět pro přidání stiskneme Submit, čímž aplikujeme změny. Tímto jsme vytvořili odchozí volání přes volbu 9 a vytáčené číslo může být jakékoliv.

Příchozí hovory zprovozníme v sekci **PBX → PBX Configuration** pod volbou **Inbounds routes**. Nás bude zajímat kolonka CID a kam máme směřovat hovor. Do kolonky CID vyplníme číslo naší linky. Nebo můžeme nechat kolonku prázdnou a tím pádem veškeré příchozí hovory se budou směřovat, kam jim zadáme cestu. V našem případě jsme nechali tuto kolonku prázdnou a hovor směřujeme na uživatele Pepu Nováka. Pro přidání této cesty zmáčkneme **Submit** a aplikujeme změny.



Obr. 39: Příchozí hovory směřované na uživatele

Nyní nás čeká napojení na **SIP** providera. V tomto návodu bude popsáno napojení na providera 802. Opět upozorňuji, že konfigurace se dle operátora může mírně lišit. Proto bychom si měli předem přesné konfigurační informace zjistit.

Podobně jako u ISDN Trunku jdeme do sekce **PBX → PBX Configuration**. Zvolíme položku Trunk a zde vybereme **add SIP Trunk**. Nás budou zajímat čtyři políčka:

- **Trunk description** – jméno SIP Trunku.
- **Trunk name** – jedinečné jméno Trunku.
- **PEER Details** – definice napojení na SIP providera
- **Register String** - registrační řetězec.

Do **PEER details** vložíme následující údaje:

Obr. 40: Konfigurace spojení na providera 802

802_číslo:heslo:802_číslo@hlas.802.cz/802_číslo

Obr. 41: Registrační řetězec

6.2.5 Tarifikace odchozích hovorů

Přejdeme-li do záložky, uvidíme záznamy všech zmeškaných, volaných, příchozích hovorů. Tyto záznamy můžeme filtrovat dle destinace, zdroje, dle časového období a také dle typu hovoru.

Date	Source	Destination	Src. Channel	Account Code	Dst. Channel	Status	Duration
2010-04-13 09:41:51	600	600	DAHDI/1-1	SIP/600-00000002	SIP/600-00000000	NO ANSWER	0
2010-04-13 09:44:15	600	600	SIP/600-00000002	DAHDI/1-1	DAHDI/1-1	NO ANSWER	0
2010-04-13 09:44:41	600	600	SIP/600-00000003	DAHDI/1-1	DAHDI/1-1	NO ANSWER	0
2010-04-13 09:45:21	600	600	SIP/600-00000004	DAHDI/1-1	DAHDI/1-1	NO ANSWER	0
2010-04-13 09:45:54	600	600	DAHDI/1-1	SIP/600-00000005	SIP/600-00000005	NO ANSWER	0
2010-04-13 09:55:00	600	600	SIP/600-00000006	SIP/802-00000007	SIP/802-00000007	NO ANSWER	0
2010-04-13 09:55:58	600	600	SIP/802-00000008	SIP/600-00000009	SIP/600-00000009	NO ANSWER	0
2010-04-13 09:57:49	600	600	SIP/600-0000000a	DAHDI/1-1	DAHDI/1-1	ANSWERED	31
2010-04-13 09:58:34	600	600	DAHDI/1-1	SIP/600-0000000b	SIP/600-0000000b	NO ANSWER	0
2010-04-13 09:58:58	600	600	SIP/600-0000000c	SIP/802-0000000d	SIP/802-0000000d	ANSWERED	72
2010-04-13 10:01:08	600	600	SIP/802-0000000e	SIP/600-0000000f	SIP/600-0000000f	ANSWERED	66
2010-04-13 10:03:03	600	600	SIP/600-00000010	DAHDI/1-1	DAHDI/1-1	NO ANSWER	0
2010-04-13 10:03:10	600	600	SIP/600-00000011	DAHDI/1-1	DAHDI/1-1	NO ANSWER	0
2010-04-13 10:04:06	600	600	SIP/600-00000012	DAHDI/1-1	DAHDI/1-1	ANSWERED	35
2010-04-13 10:12:45	500	600	SIP/500-00000013	SIP/600-00000014	SIP/600-00000014	ANSWERED	26

Obr. 42: Všechny záznamy hovorů

Pro nás důležitou záložkou v Reports je **Billing** neboli česky účtování. Proto se přesuneme tam. Vidíme, že jsme byli přeměrováni do vytváření tarifikačních plánů.

Prefix	Name	Rate	Rate Offset	Trunk	View
--------	------	------	-------------	-------	------

Obr. 43: Tarifikační plány

Pro vytvoření nového tarifikačního plánu stiskneme **Create new rate**. Otevře se nám formulář pro editaci nového plánu. Jak vidíme, stačí vyplnit jen pár políček:

- **Prefix** – tento parametr je velice důležité. Budeme třeba volat přes ISDN Trunk na mobil. Do Trunku voláme přes volbu 9. A když chceme zpoplatnit pevné linky třeba na Vysočinu, vyplníme prefix 95.
- **Name** – libovolné jméno účtovacího plánu.
- **Rate** – cena za minutu hovoru.
- **Offset** – převážně zde bude nula, protože toto pole je navýšení ceny o částku, kterou zde napíšeme
- **Trunk** – Trunk, který má být účtován. Doporučuje se dát none, protože v některých případech nemusí účtování fungovat. Spíš by se měl účtovací plán rozlišovat dle Prefixu.

Pro uložení zmáčkneme **Save**. A máme vytvořený plán. Takto můžeme postupně definovat účtování dle potřeby.

New Rate

Save Cancel

Prefix: * 95 Rate (by min): * 0.5

Name: * Pevna-ISDN Rate Offset: * 0 Trunk: * DAHDI/g0

Obr. 44: Vytvoření nového plánu

Ještě než si zobrazíme Billing Report, navštívíme nastavení **Billing Setup**. Zde si zatrhneme Trunky, které chceme zahrnout do účtování. Navíc tady můžeme nastavit výchozí účtování pro volání, které není zahrnuto v tarifikačních plánech.

Obr. 45: Nastavení účtování

Nyní se můžeme přesunout do **Billing Report**. Zde bychom měli vidět volání z Trunku, které jsme zvolili. Ale **POZOR!!!** Ve verzi elastixu 2.0 se objevuje chyba, která vznikne při automatické detekci karty. Je možné, že s jinou kartou tento problém nebude. Pokud máme kartu openvox B100P, musíme editovat konfigurační soubor v `/etc/asterisk/dahdi-channels.conf`. Přihlásíme se přes ssh do Elastixu pod root účtem. Zadáme příkaz `vi/etc/asterisk/dahdi-channels.conf`, a přepíšeme `group=0,63` na `group=0`. A tím je chyba opravena.

```

; Autogenerated by /usr/sbin/dahdi_genconf on Fri Apr 16 22:15:48 2010
; If you edit this file and execute /usr/sbin/dahdi_genconf again,
; your manual changes will be LOST.
; Dahdi Channels Configurations (chan_dahdi.conf)
;
; This is not intended to be a complete chan_dahdi.conf. Rather, it is intended
; to be #include-d by /etc/chan_dahdi.conf that will include the global settings
;
; Span 1: ZTHFC1 "HFC-S PCI A ISDN card 0 [TE] " (MASTER)
group=0,63
context=from-pstn
switchtype = euroisdn
signalling = bri_cpe_ptmp
channel => 1-2
context = default
group = 63

```

Obr. 46: Chyba v `group=0,63`

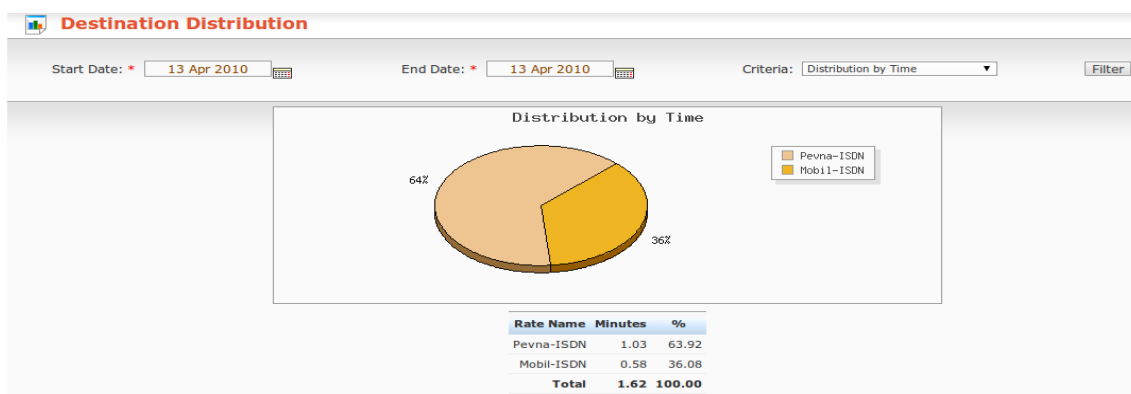
Opět jsou zde možnosti filtrace zobrazení. Můžeme si zvolit časové období odchozích volání přes Trunk. Filtrovat se dá dle volajícího, cíle volání a kanálu (Trunku). Ve sloupcích tabulky najdeme :

- **Date** – datum a čas volání
- **Rate Applied** – tarifkace, která byla uplatněna
- **Source** – uživatel, který volal
- **Destination** – cíl volání
- **Dst. Channel** – Trunk, přes který se volalo
- **Duration** – doba trvání hovoru
- **Cost** – cena hovoru
- **Summary Cost** – cena za všechna volání. Jsou zde sečteny všechny předchozí hovory.

Billing Report nám taky nabízí export do souboru ve formátu **csv**.

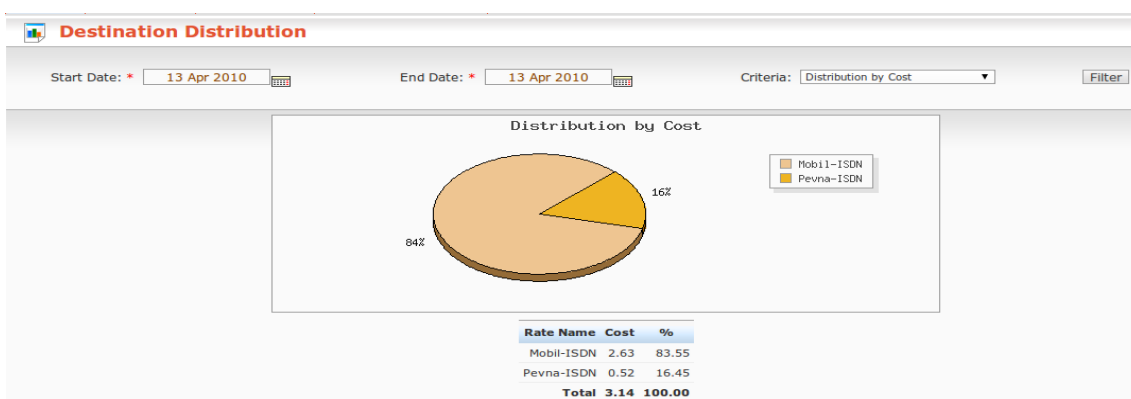
Obr. 47: Výpis hovorů s tarifací

V **Reports** → **Billing** najdeme sekci **Destination Distribution**. Pod touto volbou najdeme přehledné grafy volání jednotlivých Trunků a jejich tarifací. Při zvolení kritéria **Distribution by Time** nám graf ukáže procentuálně, kolik se provolalo minut v jednotlivých tarifikačních plánech. Samozřejmě je výběr časového horizontu.



Obr. 48: Graf volaných minut v jednotlivých tarifacích

Níže na obrázku je zobrazen graf tariface dle ceny za volání.



Obr. 49: Graf jednotlivých tarifací dle ceny

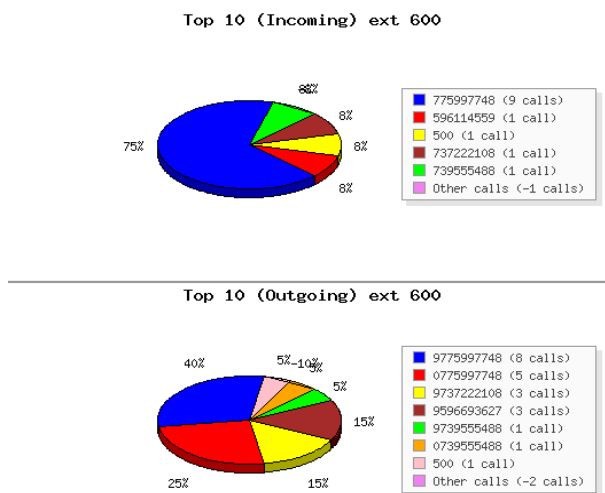
Musím zde uvést další **chybu**, která byla u verze 2.0 zjištěna. Nezobrazuje se ISDN Trunk, pokud je zvolen v Billing setup (str. 34) i SIP Trunk. Pro zobrazení musí být zaškrtnutý každý zvlášť.

Pro větší přehlednost o volání uživatelů tu je celkové shrnutí odchozích a příchozích volání. V sekci **Reports** → **Billing** → **Summary by extension** si vybereme uživatele a klikneme na **call details**.

CDR Report Channels Usage Billing ▾ Asterisk Logs Graphic Report Summary by Extension						
Summary by Extension						
<div> Start date: <input type="text" value="13 Apr 2010"/> End date: <input type="text" value="13 Apr 2010"/> </div> <div> Filter by Ext: <input type="text"/> Show </div>						
<div> Start / Previous (1 - 2 of 2) Next </div>						
Ext ▾	User	# Incoming Calls	# Outgoing Calls	Sec. Incoming Calls	Sec. Outgoing Calls	Option
500	Franta Daskocil	1	1	00h. 00m. 00s	00h. 00m. 26s	Call Details
600	Pepa Novak	12	20	00h. 01m. 32s	00h. 02m. 49s	Call Details
<div> Start / Previous (1 - 2 of 2) Next </div>						

Obr. 50: Shrnutí volání uživatelů

V grafu jsou vidět příchozí a odchozí hovory uživatele 600. Podrobně je zde zaneseno do grafu top 10 čísel a jejich počet hovorů.



Obr. 51: Top 10 odchozích a příchozích hovorů

7. Závěr

Požadavek ze strany střední telekomunikační školy byl zrealizovat IP ústřednu, která bude napojena na SIP providera a také bude napojena na veřejnou ISDN síť. Aby škola měla přehled o uskutečněném volání, požadovala, aby v systému PBX ústředny byla vyřešena tarifkace odchozích hovorů.

Nejdříve se předpokládalo, že nasazeným systémem bude opensource ústředna Trixbox. Ale setkal jsem se s problémy při integraci ovladače ISDN karty do tohoto systému. Navíc Trixbox neměl vyřešenou tarifkaci. Tudíž by se muselo hledat řešení, které by bylo kompatibilní s touto ústřednou.

Na internetu se objevil projekt, který se snaží spojit schopné opensource projekty zabývající se administrací Asterisku, nebo přidávající různá vylepšení. Místo Trixboxu byl tedy zvolen opensource PBX Elastix. Za tímto softwarem stojí španělská firma PaloSanto Solutions. Nyní se jejich ústředna nachází ve verzi 2.0. Splňuje veškeré požadavky střední telekomunikační školy. Má v sobě automatickou detekci hardwaru, proto zprovoznění ISDN karty je otázka pár minut. Navíc člověk nemusí sahat do konfiguračních souborů. Jelikož tato ústředna bude sloužit k výuce žáků této školy, prostředí Elastixu je výhodou. Účtování hovorů je řešeno jednoduše a přehledně. Správce ústředny si může vytvářet různé tarifkace a tím tak zpoplatnit uživatelům jejich volání. Proto časem může dojít k tomu, že střední škola využije toto řešení v reálném provozu.

Každý systém má bohužel své chyby a výjimkou není ani Elastix. V účtovacím systému jsem objevil dvě chyby. První se objevuje při zobrazení tarifkace hovorů (v Billing Report). Vzniká špatnou detekcí hardwaru. Chyba lze ovšem jednoduše odstranit přepsáním konfiguračního souboru. Tuto chybu zmiňuji v návodu pro realizaci. Další chyba souvisí rovněž s účtováním. Ale tentokrát se jedná o zobrazení hovorů v grafech. Při hledání příčiny jsem objevil, že nesmí být zapnuty pro grafické znázornění hovorů v Billing setup oba dva Trunky. Musíme zvolit pouze jeden a poté grafy ukazují správné hodnoty. Tyto chyby jsou nahlášený, tudíž lze předpokládat, že v příští verzi budou odstraněny.

Instalaci jsem zkoušel na třech konfiguracích. První počítač byl s minimální doporučenou konfigurací hardwaru (Pentium 3 500 Mhz, 256 Mb ram, 20 Gb Hdd). Systém jsem úspěšně nainstaloval, ale vše bylo příliš pomalé a při hovoru vznikala ozvěna. Druhý použitý počítač byl mnohem výkonnější (Intel Celeron 2,4 Ghz, 1 Gb ram, 40 Gb Hdd). Při této sestavě jsem nenarazil na žádný problém. Bohužel jsem neměl možnost otestovat, jakou zátěž by tato konfigurace snesla. Třetí testovaný počítač byl o něco málo výkonnější než druhý (Intel Celeron 3,3 Ghz, 1,5 Gb ram, 40 Gb Hdd). Urychlení bylo znát ve webovém rozhraní Elastixu. S hardwarem také souvisí další problém – s ISDN kartou. Ačkoliv karta má LED diody pro signalizaci svého provozu, ovladač pro Linux je nepodporuje. Zpočátku jsem tak měl pocit, že karta je nefunkční. Tuto skutečnost výrobce neudává a musel jsem ji sám dohledat na fóru Openvoxu.

Při konfiguraci SIP Trunku jsem zjistil jednu nepříjemnost. Pro každého SIP providera je nutná jiná konfigurace. Takže s údaji pro běžné přihlášení si člověk nevystačí a je nutné

kontaktovat poskytovatele, případně se dají získat informace na fóru telefonujeme.cz.

Veškerá popisovaná nastavení jsem vyzkoušel. Fungovaly veškeré příchozí a odchozí hovory přes Trunky i v místní síti. Lepší kvalitu hovoru mělo volání přes ISDN Trunk. Zcela nejlepší byla kvalita v rámci místní sítě. Výhoda napojení na dva Trunky je zřejmá – máme více cest, jak uskutečnit hovory.

Vlastní pobočkovou ústřednu si v dnešní době může udělat každý. Ceny hardwaru jsou na nízké úrovni a Elastix je uvolněný pod licencí GPL v.2, takže je k dispozici zadarmo. Zřízením ústředny se dá snížit rozpočet na volání na minimum.

8. Použité zdroje

8.1 Internet

[1] Sharif, B. 2010. **Elastix Without Tear**. [cit. 10.4. 2010]

Dostupné z <http://elastixconnection.com/downloads/elastix_without_tears.pdf>

[2] PauloSanto Solution. **Documentation of Elastix**. [cit. 1.3. 2010]

Dostupné z

<http://sourceforge.net/projects/elastix/files/Tutorials_Docs_Manuals/User%20Manual%200.9.alpha/Elastix_User_Manual_English_0.9-alpha.pdf/download>

[3] **Elastix**

Dostupné z <<http://www.elastix.org>>

[4] **Vi**. [cit. 15.4. 2010]

Dostupné z <<http://cs.wikipedia.org/wiki/Vi>>

[5] **Voice over Internet Protocol**. [cit. 12.11. 2009]

Dostupné z <http://cs.wikipedia.org/wiki/Voice_over_Internet_Protocol>

[6] **IP telefonie jako levnější varianta**. [cit. 12.11. 2009]

Dostupné z

<<http://tutorialy.lupa.cz/jak-nejlevneji-telefonovat/ip-telefonie-jako-levnejsi-varianta/>>

[7] Diviš, Z. 2007. **Přenosové soustavy – přednášky**. [cit. 12.11. 2009]

[8] Wailgum, T., Khudhur, P. 2008. **Abeceda VOIP**. [cit. 20.12. 2009]

Dostupné z <<http://businessworld.cz/it-infrastruktura-virtualizace/abeceda-VOIP-1555>>

[9] **Úvod do VOIP**. [cit. 30.10. 2009]

Dostupné z <<http://www.mldata.cz/uvoddoVOIP.htm>>

[10] **Jak se volá přes internet: Základy VOIP**. [cit. 30.10. 2009]

Dostupné z <<http://www.lupa.cz/clanky/jak-se-vola-pres-internet-zaklady-VOIP/>>

[11] Cesnet. **VOIP – přenos hlasu**. [cit. 15.10. 2009]

Dostupné z <<http://www.cesnet.cz/iptelefonie/VOIP-principy.html>>

[12] Khudhur, P. 2006 **Jak VOIP funguje?** [cit. 18.10. 2009]

Dostupné z

<http://www.pripojtese.cz/art_doc-2A0ABEAF4818D6BFC12571E80024FD10.html>

[13] **H.323**. [cit. 8.2. 2010]

Dostupné z <<http://en.wikipedia.org/wiki/H.323>>

[14] **H.323**. [cit. 8.2. 2010]

Dostupné z <<http://www.techabulary.com/h/h323/>>

[15] Peterka, J. 1999. **H.323**. [cit. 8.2. 2010]

- Dostupné z <<http://www.earchiv.cz/a912s200/a912s235.php3>>
- [16] **Sip** [cit. 16.3. 2010]
- Dostupné z <<https://sip.cesnet.cz/cs/protokoly/sip>>
- [17] Pužmanová, R. 2004. **Protokol Sip ve zkratce**. [cit. 16.3. 2010]
- Dostupné z <<http://www.lupa.cz/clanky/protokol-sip-ve-zkratce/>>
- [18] **Session Initiation Protocol**. [cit. 16.3. 2010]
- Dostupné z <http://cs.wikipedia.org/wiki/Session_Initiation_Protocol>
- [19] **Session Initiation Protocol**. [cit. 16.3. 2010]
- Dostupné z <http://en.wikipedia.org/wiki/Session_Initiation_Protocol>
- [20] Plánička, M. 2009 **Úvod do protokolu SIP**. [cit. 16.3. 2010]
- Dostupné z <<http://www.telegro.cz/2009/05/05/uvod-do-protokolu-sip-1-zakladni-kameny>>
- [21] Bitto, O. 2007 **Jak se volá přes internet: protokoly H.323 a Sip**. [cit. 16.3. 2010]
- Dostupné z <<http://www.lupa.cz/clanky/jak-se-vola-pres-internet-protokoly-h-323-a-sip/>>
- [22] **How To: First connection to Elastix**. [cit. 1.4. 2010]
- Dostupné z <<http://automation.binarysage.net/?p=148>>
- [23] **Telefonujeme.cz**
- Dostupné z <<http://www.telefonujeme.cz/>>
- [24] Vozňák, M. 2007 **Krok za krokem s Trixboxem**. [cit. 1.4. 2010]
- Dostupné z <<http://homel.vsb.cz/~voz29/files/TRIXBOX/Trixbox-navod.html#04>>
- [25] Zhu, J. 2008 **OpenVox B100P User manual**. [cit. 9.10. 2009]
- Dostupné z <www.openvox.com.cn/downloadsFile/B100P-User-Manual.pdf>

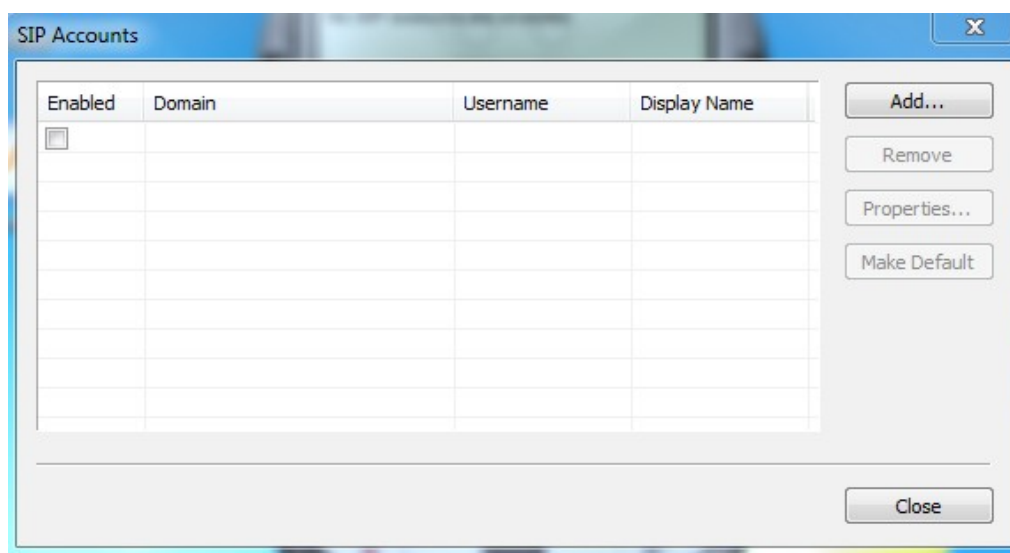
8.2 Literatura

- [26] Vozňák, M. 2008. **Voice over IP**. 1. vydání. Ostrava: VŠB-TUO Ostrava, 176 s. ISBN 978-80-248-1828-3

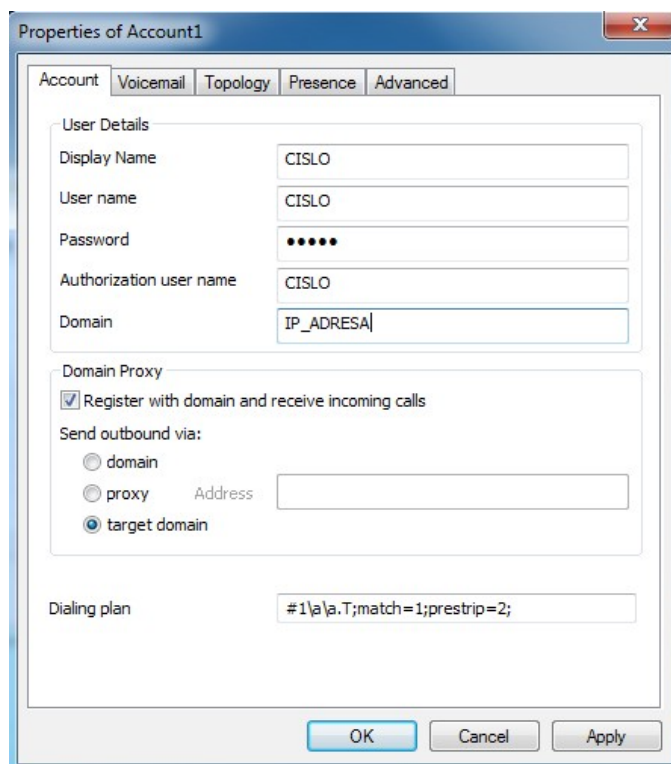
Přílohy

Konfigurace klientů

Pro počítače, kde běží Windows je k dispozici zdarma SIP klient **X-lite**. Pro napojení na Elastix je nutné vytvořit účet. Při prvním spuštění, které je vidět na obr. 52, jsme vyzváni k zadání účtu. Tedy stiskneme **Add** a vyplníme údaje podle obr. 53. Poté stačí potvrdit a klient by měl být připojený.

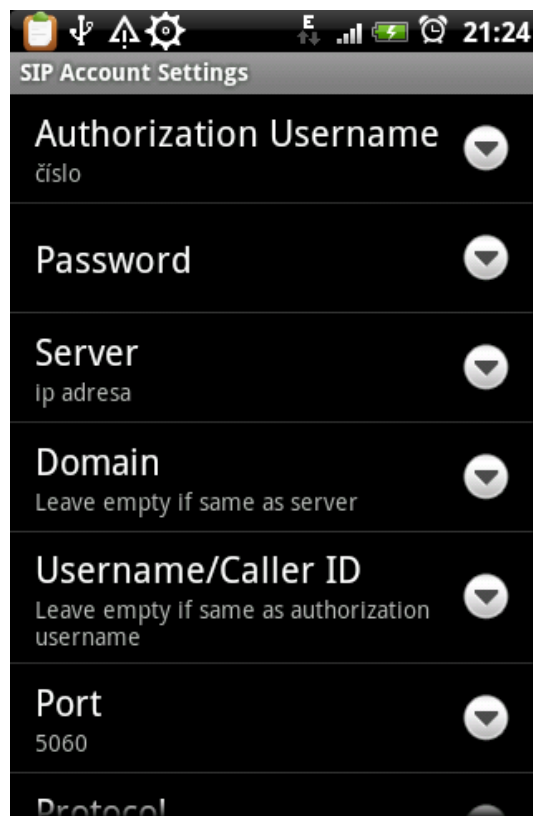


Obr. 52: Přidání účtu



Obr. 53: Přihlašovací údaje

Testovaným klientem pro mobilní platformu Android byl **Sipdroid**. Tento program je skvěle zakomponován do celého systému. Pro přenos hlasu se u Sipdroidu využívá připojení přes WiFi, UMTS, EDGE. Přes EDGE komunikaci moc nedoporučuji, jelikož je přes toto spojení špatná hovorová kvalita. Parametry v nastavení pro Elastix vidíme na obr. 53. Sekci najdeme v **Settings** → **SIP Account Settings**.



Obr. 54: Sipdroid - přidání účtu

Obsah přiloženého DVD

mat618_bakalářská_práce.pdf

Elastix-2.0.0-rc1.iso (instalační CD Elastixu)

Trunk-802.txt (konfigurace SIP Trunku providera 802)

Složky s tabulkami,obrázky a zdrojové soubory dia